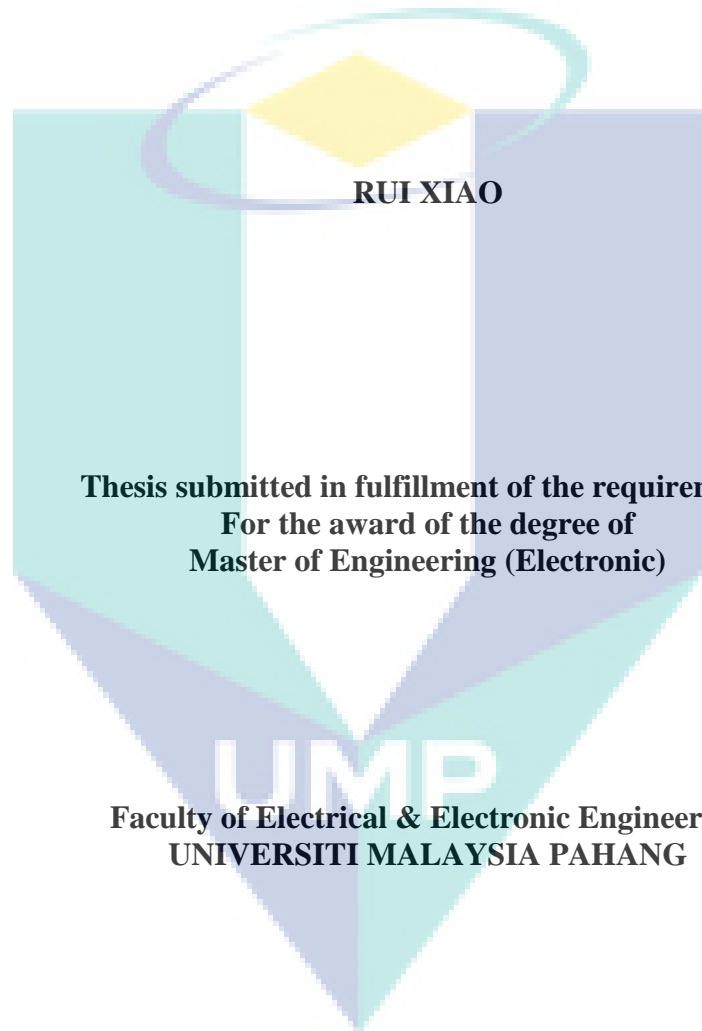


**LANE MARKINGS DETECTION BASED ON  
E-MAXIMA TRANSFORMATION AND IMPROVED HOUGH**



**Thesis submitted in fulfillment of the requirements  
For the award of the degree of  
Master of Engineering (Electronic)**

**UMP**  
**Faculty of Electrical & Electronic Engineering**  
**UNIVERSITI MALAYSIA PAHANG**

**MARCH 2013**

### **SUPERVISOR'S DECLARATION**

I hereby declare that I have checked this thesis and in my opinion this thesis is satisfactory in terms of scope and quality for the award of the degree of Master of Engineering (Electronic).

Signature :

Name of Supervisor : DR. KAMARUL HAWARI BIN GHAZALI

Position : SENIOR LECTURER

FACULTY OF ELECTRICAL & ELECTRONIC ENGINEERING,  
UNIVERSITI MALAYSIA PAHANG

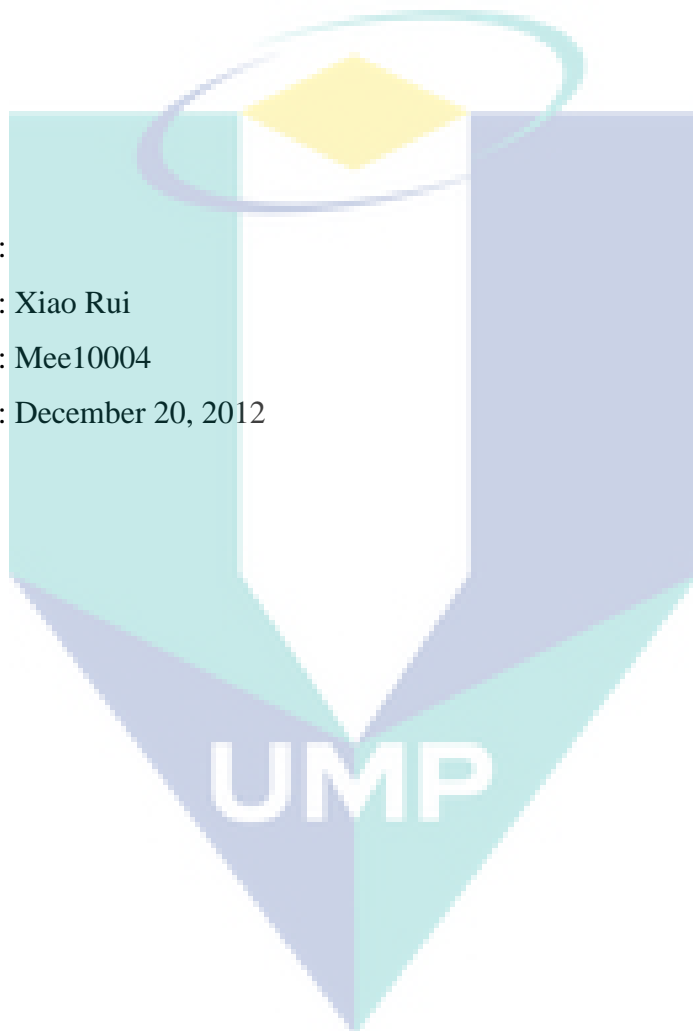
Date : December 20, 2012

**UMP**

### STUDENT'S DECLARATION

I hereby declare that the work in this thesis is my own except for quotations and summaries which have been duly acknowledged. The thesis has not been accepted for any degree and is not concurrently submitted for award of other degree.

Signature :  
Name : Xiao Rui  
ID Number : Mee10004  
Date : December 20, 2012

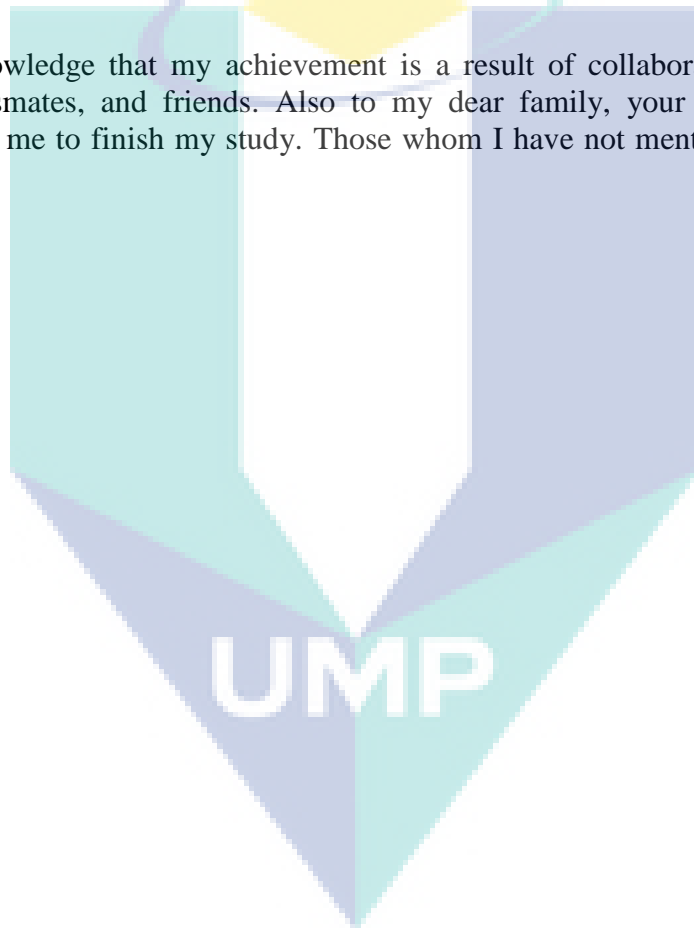


## ACKNOWLEDGEMENTS

I would like to express my most sincere gratitude to my supervisor Dr. Kamarul Hawari bin Ghazali, for his dedicated support, professional guidance, and showing me how research is conducted.

My deepest gratitude should be forwarded to Vice-Chancellor of University Malaysia Pahang (UMP), Professor Dato' Dr. Daing Nasir Ibrahim for granting me the GRS scholarship.

I acknowledge that my achievement is a result of collaborating efforts from my lecturers, classmates, and friends. Also to my dear family, your ultimate concern and support enable me to finish my study. Those whom I have not mentioned above I wish to thank you all.



## ABSTRACT

According to Malaysians Unite for Road Safety (MUFORS) online survey, human error, for example, improper vehicle deviation or unintentional lane change is one of the main causes of traffic accident. Lane shift in traffic can be complex and dangerous. This study aims at developing a fast, low-cost, and sophisticated system with the ability to detect unexpected lane changes that may reduce the probability of a vehicle straying out of lane. Various road models to identify the lanes have been explored including straight-line, B-snack, linear-parabolic model, and deformable model. Most lane models, either simple or lack of flexibility or complex, may cause heavy computation in processing the time needed. The feature of roadway has certain degree of curvature and constraints, for instance, no sudden road turn is the design for road safety driving. A short segment of a long curve with a relatively low curvature is approximated as a straight line, based on this point, the important contribution of this thesis presents a lane detection algorithm using E-MAXIMA transformation and improved Hough transform which is the algorithm with great efficiency, high robustness and also at low cost to detect road lane markings. First of all, the region of interest from input image to reduce the searching space is defined; then the image into near field-of-view and the far field-of-view is divided. In the near field-of-view, Hough transform will be applied to detect lane markers after image noise filtering and lane features extraction by E-MAXIMA. The experimental results based on collected video data under complex illumination conditions had proved that the proposed algorithm is able to detect the road lane marking efficiently achieving a correction rate of 95.33%. The process time on average is 32 ms/f, namely every second can deal with 31.25 frames that demonstrate superior and robust results compared to other existing methods. To conclude, the work done in this thesis may apply to autonomous driving navigation and driving security assistance. The potential of such a system is further linked to the system with the vehicles' turn signal, whereby the system will be able to detect an unintentional drift out of the lane.

## ABSTRAK

Menurut kaji selidik oleh Malaysia Unit for Road Safety (MUFORS), kesilapan manusia seperti kereta terkeluar dari lorong atau teralih dari lorong asal merupakan punca utama berlakunya kemalangan jalan raya. Teralih kenderaan semasa memandu adalah satu perbuatan yang bahaya dan kompleks. Kajian ini bertujuan membina satu sistem yang cepat, canggih, dan murah serta berkeupayaan untuk mengesan perubahan lorong yang tidak dijangka. Sistem ini dapat mengurangkan kebarangkalian kenderaan terkeluar daripada lorong. Pelbagai model jalan telah dikenalpasti dalam kajian ini termasuk garis lurus, B-snack, model parabola-lurus, dan model bolehubah bentuk. Kebanyakan model tidak terkira yang mudah atau kurang fleksibiliti atau kompleks akan mengambil masa yang lama dalam proses pengiraan oleh perisian. Biasanya, semasa merekabentuk jalan untuk memandu dengan selamat, ciri-ciri jalan mempunyai tahap kelengkungan dan kekangan tertentu dan tindak terdapat belokan secara tiba-tiba. Satu segmen yang pendek bagi satu lengkungan yang panjang boleh diandaikan sebagai garisan yang lurus. Berpanduan kepada andaian tersebut, thesis ini dapat memberi sumbangan dengan mengemukakan satu algoritma pengesanan lorong. Algoritma pengesanan lorong tersebut dibentuk dengan menggunakan tranformasi E-MAXIMA dan menambahbaik trasformasi Hough. Tranformasi Hough merupakan algoritma yang cekap, mempunyai fungsi mengasingkan gangguan dan murah untuk mengesan tanda-tanda lorong. Pada permulaan, tentukan dan iktirafkan kawasan yang diminati dengan memasukan imej bagi mengurangkan skop pencarian. Selepas itu, bahagikan kawasan imej kepada pandangan dekat dan jauh. Bagi kawasan pandangan dekat, Hough transform akan digunakan untuk mengesan tanda-tanda di jalan selepas gangguan ditapis dan diasingkan oleh E-MAXIMA. Keputusan eksperimen yang berdasarkan kepada data dari video, yang dikumpul bawah keadaan illuminasi yang kompleks, telah membuktikan algoritma yang dicadang boleh mengesan tanda lorong jalan raya dengan cekap sehingga mencapai kadar pembetulan sebanyak 95.33 %. Purata masa proses ialah 32ms/f, iaitu algoritma ini boleh menangani 31.25 bingkai tayangan setiap saat dan telah membuktikan bahawa keputusan yang didapati dengan cara ini adalah lebih baik dan tepat berbanding dengan cara lain. Kesimpulannya, hasil penemuan tesis ini boleh diaplikasikan kepada navigasi autonomi memandu dan bantuan memandu secara selamat. Sistem ini berpotensi bergabung dengan sistem isyarat perubahan belokkan kenderaan, dimana sistem ini dapat mengesan terkeluarnya kenderaan dari lorong asal. Penyelidikan selanjutnya adalah untuk megaitkan system tersebut dengan penukaran isyarat kenderaan dan kajian tentang perhubungan geometri pada jalan yang sangat tidak rata supaya mengurangkan kemalangan jalan raya.

## TABLE OF CONTENTS

<b>SUPERVISOR’S DECLARATION</b>	I
<b>STUDENT’S DECLARATION</b>	II
<b>ACKNOWLEDGEMENTS</b>	III
<b>ABSTRACT</b>	IV
<b>ABSTRAK</b>	V
<b>TABLE OF CONTENTS</b>	VI
<b>LIST OF TABLES</b>	X
<b>LIST OF FIGURES</b>	XI
<b>LIST OF ABBREVIATIONS</b>	XIII
 <b>CHAPTER 1 INTRODUCTION</b>	 1
1.1 INTRODUCTION	1
1.2 MOTIVATION	2
1.3 PROBLEM STATEMENT	3
1.4 OBJECTIVES OF THIS STUDY	4
1.5 SCOPE OF THIS STUDY	4
1.6 STRUCTURE OF THE THESIS	5
 <b>CHAPTER 2 LITERATURE REVIEW</b>	 6
2.1 INTRODUCTION	6
2.2 LD SYSTEM BASED ON MONOCULAR AND BINOCULARS VISION	7
2.3 THE LANE-REGION SEGMENTATION BASED METHODS	7

2.3.1	Luminance Thresholding	8
2.3.2	Texture anisotropy	9
2.3.3	The Region growing method	10
2.3.4	The Watershed Transform	11
2.4	THE FEATURE-BASED METHODS	12
2.4.1	RGB color model	12
2.4.2	Hue-Saturation-Intensity model	13
2.4.3	Edge detection	15
2.4.4	Ridge detection	15
2.4.5	The Inverse Perspective Mapping	16
2.5	THE MODEL-BASED METHODS	18
2.5.1	A linear-parabolic lane model	19
2.5.2	Likelihood Of Image Shape	19
2.5.3	Non Uniform B-Spline	20
2.5.4	B-Snake based lane model	21
2.5.5	The RANSAC algorithm	21
2.5.6	The Hough Transform	22
2.5.7	Other extensions to the HT	24
2.6	CONCLUSION	24
<b>CHAPTER 3</b>	<b>METHODOLOGY</b>	<b>26</b>
3.1	DATA ACQUISITION	27
3.1.1	Image processing platform	28
3.1.2	Python and MATLAB	28
3.1.3	Data collection	29
3.2	REGION OF INTEREST SELECTION	30
3.3	FEATURE EXTRACTION BASED ON E-MAXIMA TRANSFORMATION	31



3.4	EDGE DETECTION TECHNOLOGIES	33
3.4.1	Sobel operator	35
3.4.2	Roberts cross operator	37
3.4.3	Prewitt operator	39
3.4.4	Canny Edge Detector	39
3.5	LANE MODEL ANALYSIS	41
3.6	LANE DETECTION TECHNIQUE	44
3.6.1	The algorithm1: Improved Hough based on prior-knowledge	46
3.6.2	The algorithm2: PPHT	48
3.6.3	The algorithm 3: RANSAC	50
3.7	LANE MARKING EXTRACTION	52
3.8	LANE DEPARTURE DECISION	54
3.9	THE ERROR-CATCHING MECHANISM	56
3.10	CONCLUSION	58
<b>CHAPTER 4</b>	<b>RESULT AND DISCUSSION</b>	<b>59</b>
4.1	THE EXTENDED-MAXIMA EFFECT	59
4.2	EXTRACTION OF LANE EDGES	63
4.2.1	Threshold evaluation	63
4.2.2	Edge connectivity analysis.	65
4.2.3	Anti-noise capability analysis	67
4.2.4	Comparison of edge detection efficiency	67
4.3	LINE DETECTION TECHNIQUE	69
4.3.1	Behavior of the line fitting	69
4.4	EXPERIMENT OF LANE DEPARTURE DECISION	71

4.5	COMPARISON OF THE SHT, IMPROVED HOUGH AND PPHT.	76
4.6	PERFORMANCE EVALUATION OF IMPROVED HOUGH AND RANSAC	77
4.7	EXPERIMENT ON DETECTION ERROR APPEARANCE	80
4.8	CONCLUSION	81
<b>CHAPTER 5</b>	<b>CONCLUSION</b>	83
5.1	Future Work	85
<b>REFERENCES</b>		86
<b>LIST OF PUBLICATIONS</b>		93

A large, faint watermark of the UMP logo is centered on the page. It features a shield-like shape composed of four colored triangles (teal, light blue, yellow, and light purple) meeting at a central point. Below the shield, the letters "UMP" are written in a bold, white, sans-serif font.

UMP

## LIST OF TABLES

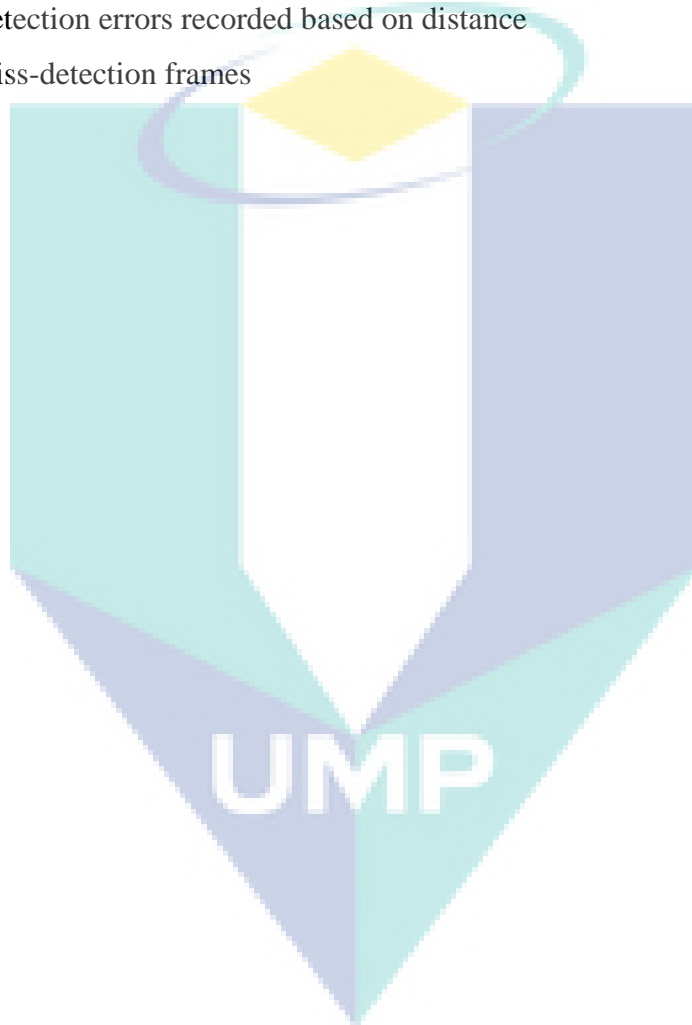
<b>Table No.</b>	<b>Title</b>	<b>page</b>
3.1	Hardware specifications	28
3.2	Videos specifications	28
3.3	The outdoor pre-conditions of Data collection	30
3.4	Minimum radius for road design	42
4.1	The connected components in edge map	66
4.2	Comparison of classical derivative operators on average running time	68
4.3	The maximum voting result from the Hough accumulator	69
4.4	The result of detected lines	70
4.5	The relationship between lane markings and deviation angles	75
4.6	The computational efficiency for lane marking Images	76
4.7	The accurate rate of the Hough algorithms	77
4.8	Accuracy rates of the Lane Detection Technologies	78

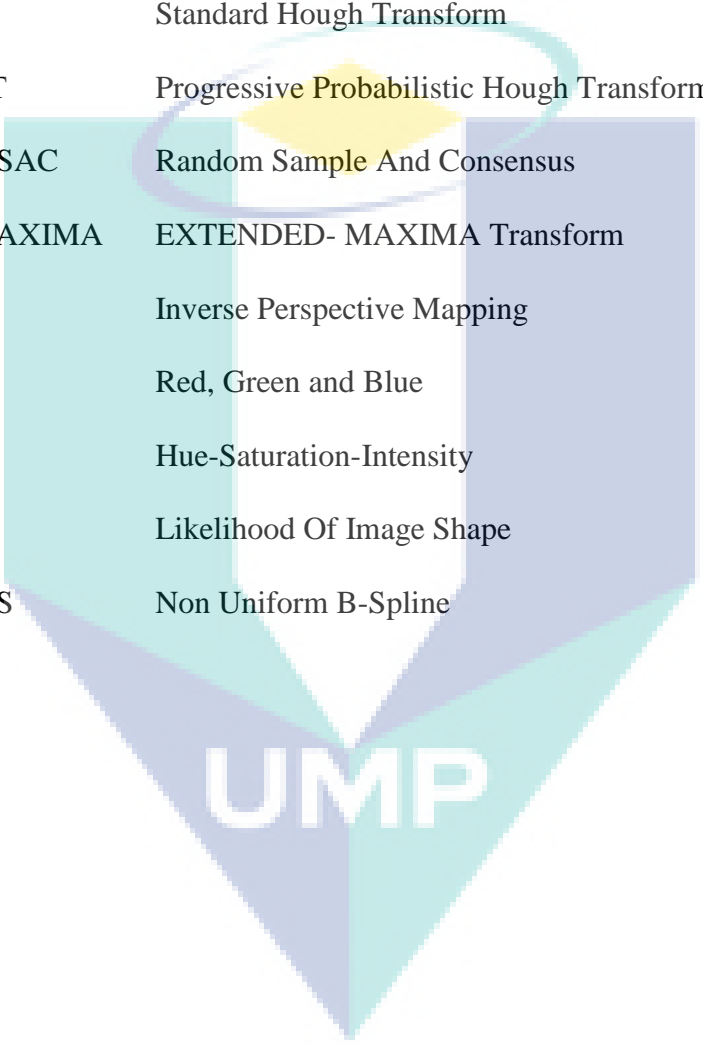
UMP

## LIST OF FIGURES

Figure No.	Title	page
2.1	The HSI color space	14
2.2	Lane markings resemble mountains	16
2.3	Inverse perspective mapping	17
2.4	Modeling for road-lane-curvature estimation	20
2.5:	An example of ROI	23
3.1	The main algorithm flow-chart	26
3.2	A representation of data acquisition	27
3.3	An example of ROI selection	31
3.4	A cross section in road-surface image	33
3.5	Distribution of pixels value in a cross section.	33
3.6	The difference between using a straight line and a curve model	43
3.7	An example of using straight lane model on curve road	43
3.8	Mapping of one line to the Hough space	45
3.9.	Transformation of points to lines in the Hough space	45
3.10	A profiled lane marking illustration	53
3.11	The driving positios of vichiel	55
3.12	Distance h between right and left lane marking	57
4.1	Process of E-MAXIMA effect evaluation	59
4.2	Examples of edge detection after E-MAXIMA.	60
4.3	Applying E-MAXIMA transformation on low level of illumination	62
4.4	The edge map resulted by different threshold values	64
4.5	The neighborhood structure components	65
4.6	An example of linear edge markings	66
4.7	A histogram compares the edge maps resulted from operators.	67
4.8	Average computation time using the four type of edge operators	68
4.9	The plotted pixels in the target image	71
4.10	The detection results on different roads	72

4.11	Lane departure parameters extraction	73
4.12	Computational performance in the experiment	78
4.13	Detection effect implemented by RANSAC algorithm	79
4.14	Detection errors recorded based on Theta value	80
4.15	Detection errors recorded based on Rho value	80
4.16	Detection errors recorded based on distance	81
4.17	Miss-detection frames	81



**LIST OF ABBREVIATIONS**

LD	Lane Detection
ROI	Region of interesting
SHT	Standard Hough Transform
PPHT	Progressive Probabilistic Hough Transform
RANSAC	Random Sample And Consensus
E- MAXIMA	EXTENDED- MAXIMA Transform
IPM	Inverse Perspective Mapping
RGB	Red, Green and Blue
HSI	Hue-Saturation-Intensity
LOIS	Likelihood Of Image Shape
NUBS	Non Uniform B-Spline

## CHAPTER 1

### INTRODUCTION

#### 1.1 INTRODUCTION

Image processing has been found in many applications in the electronic industries. The field of autonomous vehicles and driver support has received attention. Lane detection technologies are the branch of intelligent driver assistance system used to warn a driver when the vehicle begins to veer out of its lane. The purpose is to lessen possibility of traffic accidents, to monitor the position of a car effectively and to contribute to further development of autonomous navigation technology.

The subject in this thesis is about algorithms of lane extraction, detection and location which are based on digital video sensor. It is relatively advantageous than other type of sensors like Radar and infrared. It has a lower cost, more mature applications, and can provided rich visualized information. However those road information provided by video is disorderly and not standardized, thereby a number of extra approaches are needed to extract the specific features of lane markings, which is a crucial subject. A reliable detection algorithm has been keeping exploring in research area.

This project presents simulations to analyze the driving control for autonomous vehicle using image processing technology to detect road lane. The approach provided here is based on E-MAXIMA and improved Hough Transform to extract the features of structured roads. The near field-of-view scope adopts a straight line model to accelerate the speed of data calculation and to find the fitting line. Prior-knowledge is used in lane finding process to efficiently decrease Hough space efficiently, thus enhancing its robustness by

improving the processing speed. The algorithm gave a good result in detecting straight and smooth curvature lane on the highway even when the lane was affected by shadow.

Furthermore, RANSAC and Progressive Probabilistic Hough Transform (PPHT) are used as alternative technologies to implement lines detection function, for comparing purpose. The advantages and disadvantages were explored by observing the algorithms performance under similar experimental conditions. The methods have been tested on collected video data. Experimental results demonstrated that the efficiency and robustness of the purposed algorithms is ideal.

The important contribution of this study is the development of vehicle lane detection and tracking algorithm based on E-MAXIMA and Hough transform. Major consideration in this research includes resolving speedy detection of lane markings and locating accurate vehicle position. To develop a high precision, simple computation and strong adaptability algorithm are applied for real-time requirements.

## **1.2 MOTIVATION**

Malaysia is paying a heavy price due to road accidents, and the cost to the economy 2010 year was about RM9.3bil (Malaysia Road Safety Department, 2010). Recent statistics shows that the number of fatalities in Malaysia has increased to 6,872 deaths in last year. Bukit Aman Internal Security and Public Order Department revealed that the total number of road accidents has increased to 414,421(MUFORS, 2011). In a separate online survey by the Malaysians Unit for Road Safety (MUFORS), 61.6% of the respondents believed that human error, for instance, improper vehicle deviation or unintentional lane change is one of the main causes of road carnages.

Changing lanes in traffic can be complex and dangerous, and such detection system may reduce the probability of a vehicle straying out of lane. To prevent a vehicle veering out of lane, technique for driving assistance is vitally important, particularly the Lane Markings Detection System which has a significant market potential and high practical



value. The U.S. National Highway Traffic Safety Administration (NHTSA) began studying whether to mandate lane detection systems on automobiles (Rothschild, 2012). The study of lane detection technique is moving towards comprehensiveness, generalization, and digitalization.

### 1.3 PROBLEM STATEMENT

Lane detection is commonly used to determine some geometric parameters including the shape and width of the lane region, the lane markings position, and analysis of derivation angle. Existing techniques in the study of lane detection technology have a diversified angle in analysis and possess a variety of disadvantages. The complicated road condition experimental results have been far from ideal. To build up a systematic criteria and comprehensive solution for safety driving is still a long way to go. The main focus is how to improve the efficiency and accuracy of an algorithm.

The proposed approaches aim to confirm high stability in noise conditions, change of illumination and outside noise from other vehicles. There are several technical challenges to road detection which is needed to be understood, analyzed and solved:

- (i). On the side of the road shadows projected by trees, buildings, bridges to the surface can interfuse edges information and the irregular form of blot or reflections of wet road can also create non-road edges. In these cases, it would be difficult to eliminate interfered edge information and well preserved useful edge information.
- (ii). Lane markings can be frayed or smudged with years since edges will be faded. The bumping and shaking of vehicles or poor weather can produce blurred images. Moving vehicles, pedestrians or obstacles appear on the road will conceal part of the markings. How can we distinguish the lane markings from the obstacles?

- (iii). Review on previous research, the critical technical difficulties of lane detection is conflict between robustness and the real-time applicability. In point of robustness, algorithms should be able to work under variety illumination and weather that is resulted by the complicated computation process. On the other hand, the real-time applicability request algorithms are able to complete detection process less than spans of one second. Thus, in order to satisfy the need of real-time requirement and robustness still needs lots of research work.

#### **1.4 OBJECTIVES OF THIS STUDY**

The objective is to develop and design an algorithm for efficient and effective road lane detection which can increase the comfort and safety of traffic participants. In order to minimize the accidents, the main cause of collision, that is, driver errors, is addressed. The present study will analyze the situation and relevant departure information. This research is implemented on autonomous navigation and assistance on the visually impaired can decrease the number of traffic accidents. The specific objectives of this research are:

- (i). To detect road lanes in different environmental conditions in the presence of noise on the road.
- (ii). To prove the optimal edge detection operators for lane detection purpose and to obtain the best automatic threshold under different lighting conditions.
- (iii). Improve performance of standard Hough and verify it by mathematical operations which means a simulation should be implemented.

#### **1.5 SCOPE OF THE STUDY**

The central theme of this thesis is to design and implement a reliable solution for real time lane detection. This research is based on a computer vision by a single video camera as an input sensor. Autonomous vehicles and driving assistance based on computer vision is a comprehensive topic. The scope of this research covers the following items:

- (i). Achieve recognition by extracting specific features of lane markings painted on the road surface with recognizable painted lane markings via video sensor.
- (ii). By analyzing the relevance of the lane positions between the adjacent frames, we can use the information obtained to guide the detection of the next frame.
- (iii). Find a lane model that explains the features found in the current image and the previous frames.
- (iv). Detect and track the vehicle's location and counting distance between vehicle and lane marks.
- (v). Determine and analyze the “derivation behavior” of the host vehicle.

## **1.6 STRUCTURE OF THE THESIS**

Chapter 1 provides a brief introduction and research background of the lane detection system. This is followed by the problem statement, objective of the system and scope of the study.

Chapter 2 reviews the relevant literature, discusses the current adopted widely technologies and offers a classification based on the different computation techniques or different objects, such as color, texture, edges or shape.

Chapter 3 introduces the image pre-processing technologies used in this Lane Detection system (LD) and illuminates working theory of improved Hough and RANSAC algorithms. The lane detection process of the designed LD system will be presented and technical reasoning and feasibility of adapting straight lane model will be discussed.

Chapter 4 describes the experiments conducted in the designed LD system. The comparative results of different algorithms and analysis are presented. The final chapter summarizes the limitations of the system and suggestions for future research will be made.

## CHAPTER 2

### LITERATURE REVIEW

#### 2.1 INTRODUCTION

Extensive techniques have been developed to detect lane markings in the related field of autonomous vehicles and driver support technologies. This chapter presents an overview of the foremost techniques used by these researchers. Image-processing methods used in traffic applications will be reviewed and then group them according to the type of technologies applied. An evaluation of the advantages and disadvantages from general needs will be conducted. The evaluation result will help position our work in the context of previous research and creating 'research space'.

Till now a large number of Lane Detection systems have been developed by diverse types of sensors, for example: radar and infrared sensors, inductive loop, and microwave detectors. Comparing with video sensors, serious drawbacks of those sensors are high installation and maintenance expense. On the contrary, video sensors offer a relatively inexpensive cost, as well as slight traffic disruption.

The commercial use of video sensors is increasing. Our work in this research will be concentrated on developing LD algorithms via video sensors. Generally the LD system based on video sensors can be divided by monocular vision (one eye camera) and binocular vision (two-eye camera). Moreover, according to different detection algorithms, current research can be categorized under three types: a) the Lane-region based methods, b) the feature-driven methods and c) the model-driven methods (Obradović, Konjović, Pap et al., 2012), (Y. Fan, Zhang, Li, Zhang, et al., 2011), (Kastrinaki, Zervakis, & Kalaitzakis et al.,

2003). The lane-region segmentation based method can be also modeled as an image segmentation issue which is the process of partitioning an image into road and non-road segments based on particular features. The feature-driven methods estimate the road by intensity, color, texture, edge strength or ridge. The other model-driven methods actually can be considered as a matching process that is to compare deformable template or parameterized shapes with observed images.

## **2.2 LD SYSTEM BASED ON MONOCULAR AND BINOCULARS VISION**

Commonly, current active vision-based LD system can be grouped by different number of video sensors, e.g. monocular and binocular vision. Monocular vision system in W. He et al., (2011), Schreiber, Alefs, & Clabian et al. (2005) has one video camera sensor fixed on the rear mirror to collect road surface information. Binocular vision (Lipski et al., 2008) (Rios Cabrera, Tuytelaars, & Van Gool, 2011) had fixed two cameras in the front-right and front-left side of the vehicle that can be integrated to take pictures from two sides of lane edge at short ranges. By this way, camera shooting towards the road and image can be captured quickly, at the same time it eliminates useless background information during image acquisition phrase.

The limitations of such system based on two/multi-camera sensors need to process huge amounts of real-time images. It will request complex hardware support, otherwise it will slow down the computation speed, raise the complexity of process algorithm; furthermore, synchronization of analyzing results in each frame from different camera sensors would be very difficult, sometimes leading to miss detection.

## **2.3 THE LANE-REGION SEGMENTATION BASED METHODS**

The lane-region segmentation based methods can be defined as a graph partitioning technique, which subdivides an image into its constituent road and/or non-road segments based on the road regions with similar attributes (Ben Romdhane, Hammami, & Ben-Abdallah, 2011) (Kastrinaki et al., 2003). It is typically used to extract the lane positions.

The segmentation results can simplify the content of images to make road information more meaningful and easier to analyze. Several lane-region analysis methods are discussed in this subsection which are based on basic properties or detect the lane with the changing intensity distribution along the region of a lane such as the thresholding of luminance, color component, and watersheds. Another category of segmentation is accomplished via flood-fill the road region using predefined criteria, such as region growing method.

### 2.3.1 Luminance Thresholding

Several analytic algorithms to the setting of a luminance threshold have been proposed. The process is to create binary images depending on the value of pixels. Individual pixels in an image are marked as two dominant models, object and background pixels. Threshold  $T$  extracts the objects from the background. Any point  $(x,y)$  in the image at which  $f(x,y) > T$  is called an object point; otherwise their value is smaller than  $T$  which is labeled as a background point (Gonzalez & Woods, 2008). The segmented image,  $g(x,y)$ , is given by

$$g(x,y) = \begin{cases} 1 & \text{if } f(x,y) > T \\ 0 & \text{if } f(x,y) \leq T \end{cases}$$

As noted in the previous equation, under the condition that the intensity distribution of objects and background pixels are readily distinguishable, a single (global) threshold can get a good segmentation result. Besides, Otsu's method is very common to select the optimum threshold value. Liu et al. (2012) have reported that the Otsu method is better than the global thresholding technique among their experimental results. Another approach to luminance threshold selection is to find the smallest point of the histogram between its bimodal peaks. Zhang & Wu (2009) used two histogram images to create two different calculation directions, horizontal and vertical histogram. The original image is segmented twice by row and column thresholds separately.

When a different threshold is used for different regions in the image it is called the adaptive threshold. This paper (Z. Li, Cai, Xie et al., 2012) is to describe a method using an adaptive threshold approach for real-time automated extraction of the road markings.

The method of applied multiple threshold segmentation (H. Wang & Shao, 2011) is used instead of single threshold. All parts of lane marker points are collected by using a set of consecutive thresholds. Consecutive threshold segmentation uses multiple thresholds and collects useful information from each segmented image. A white line segment will be selected as part of lane marker points if its length satisfies certain length range. Using consecutive threshold method can separate lanes and background but less flexible for complex data. Under uneven illumination conditions, threshold selection would be a major challenge.

Usually a successful segmentation is highly dependent on the choice of thresholds. Global (single) thresholding works on different scope of image. It can be expected to be successful in highly contrast environments. Its advantages are simple and the applicability is strong. Those pictures with even illuminance using global thresholding can achieve a good segmentation result. Its disadvantages are low capacity of resistance noise and less sensitive to the intensity variation of the grey image. Adaptive thresholding, also called local thresholding, has improved anti-noise. However, segmentation errors can be occurred when neighborhood subdivision is filled by objects or background pixels. Multiple or consecutive thresholds are applicable to complex background image, which increases the flexibility of thresholds selection at the same time increases the complexity of algorithms. In general, multilevel thresholding is less stable, mostly because it is very difficult to determine thresholds that are adequately separated objects.

### **2.3.2 Texture anisotropy**

Existent approaches to texture, instead of gray value, can be a useful feature for lane region segmentation. The texture of the road is normally smoother than that of the

environment (Jeong & Nedeveschi, 2005). The proposed method is carried out on the sub images by a composed gray and texture based feature vector. The gray feature vector works in the pre-classification phase, and the texture feature vector is applied after the classification is performed. The proposed method using in this paper is dexterous to lane detection but the weakness shows in the strong variant environment. The feature vector of the object and the background become similar when the entire image, for example, dark or bright. In this case, the miss detection of the road region can easily occur (Kang, Kidono, Naito, & Ninomiya, 2008). Filter banks are used to measure texture properties in the image. Each pixel records the local response of a different filter. Pixels with similar responses indicate regions with similar textures.

Since the road surface is smooth and differs significantly from objects (vehicles) and background, texture-based segmentation is possible to identify the road region area from other background. However, the detection result often shows instable results that were caused by too much segmentation in objects of interest where similar textures seemed to end up in different areas. Moreover, in highway scenario, texture of one lane does not have much difference from the near or next lane. Practically, it is not very feasible to measure texture properties in highway scene.

### **2.3.3 The region growing method**

The Region-growing method is to divide a digital image into multiple portions depending on similar properties such as pixel intensity, gray level texture, or color. The first step is to select initial seed points from the lane-region. Then the initial regions are grown from these initial seeds which are determined by the definite location of regions. It is an iterative process, that is, keep searching for similarity from neighboring pixels until road boundaries are allocated to the region.

According to Song & Civco (2004), the image is classified into two groups of categories: a road and a non-road group. The road group image was segmented into



geometrically homogeneous objects using a region growing technique based on shape information. Amo, Martinez, & Torre, (2006) have presented the applicability of a combined approach consisting of region growing and competition to extract roads. An initial simple model is deformed by using region growing techniques to obtain a rough road approximation. The extraction result is able to obtain the road centerline and the road sides.

The trouble with the region growing method is that road gaps caused by shadow or obscuring land features can stop the region growing early, producing erroneous segmentation results. For lane markings detection purpose, the method is not explicit to localized target objects, instead, it takes a devious way to acquire lane markings by seeking road region first, and extra computation time would be required.

#### **2.3.4 The Watershed Transform**

One of the important tools in image segmentation is the watershed transformation. The approach in X. Yu et al. (1992) was based on mathematical morphological segmentation to locate the lane edges in the image. The watershed is given by considering the graph of image as a topographic surface. By applying the watershed of the gradient image a mask is created. Then depending on the filtered mask the watershed finds road edges. This technique has the advantage of not requiring any threshold for the gradient magnitudes and no geometrical model is necessary. It has the disadvantage of not imposing any global constraints on the lane edge shapes.

Beucher & Bilodeau (1994) had worked on road segmentation and obstacle detection based on watersheds. Their technique consists of applying a temporal filter for noise reduction, followed by edge detection and watershed segmentation. Such methods demand a relatively high computational cost, and the resulting road boundaries are typically jagged.

## 2.4 THE FEATURE-BASED METHODS

The feature-based methods search the lanes from the road images by extracting the low-level features. Those low-level features of road are basic features that can be extracted automatically from an image without any shape information (Nixon & Aguado, 2008)

The most noticeable features supporting the estimation of the road course are intensity, color, texture, edge strength, edge direction and height-over-ground (Franks, Loose et al. 2007). Some road detection approaches based on color features will be reviewed firstly. Those approaches are marked image pixels into road markings based on particular color features. A color space is a mathematical representation of a set of colors. The three most popular color models are a) RGB (used in computer graphics); b) YUV, or YCbCr used in video systems; and c) CMYK used in color printing. However, none of these color spaces are directly related to the intuitive notions of hue, saturation, and brightness. This resulted in developing other models, such as HSI and HSV, to simplify programming, processing, and manipulation.

### 2.4.1 RGB color model

The use of color in image processing is motivated by one principal factor: color is a powerful descriptor that often simplifies objects identification and extraction from a background (Gonzalez & Woods, 2008). Researchers have used contrasting color and intensity on the road surface as important and preferred analytical elements to distinguish lane markings on the road surface. In the RGB color model, the features are defined by the spectral components of the illumination at the red, green and blue bands. At each pixel, the classification can be performed directly on the (R, G, B) bands.

Chiu & Lin (2005) had made some attempts in vision based lane detection by RGB color model. The researcher has discovered a threshold using statistical method in a color image. This is used to distinguish possible lane boundary from the road. Generally most of the road is near the color of dark gray. Although it is not exactly gray, it is still in a certain

range. Under the RGB color model, a pixel said to be gray is with the property  $V_R = V_G = V_B$ . And, it is nearly white when  $V$  approaches 255 and is nearly black when approaches 0. However, on the real road, it is not exactly the color of gray when it is repaired or adding some other colors due to the reflection of light. Furthermore, the color of a scene may vary with time. All  $R$ ,  $G$  and  $B$  values are required to distinguish the different thresholds with time and environmental change.

The determination of the threshold value would be a difficult issue when using the color feature. For general road, the gray level value of the road is 100-130, and its shadow or sunlight would be changed in various scales. The apparent color of an object is neither static nor constant. Under those factors of affection, such kind of systems might get inconsistent detection result.

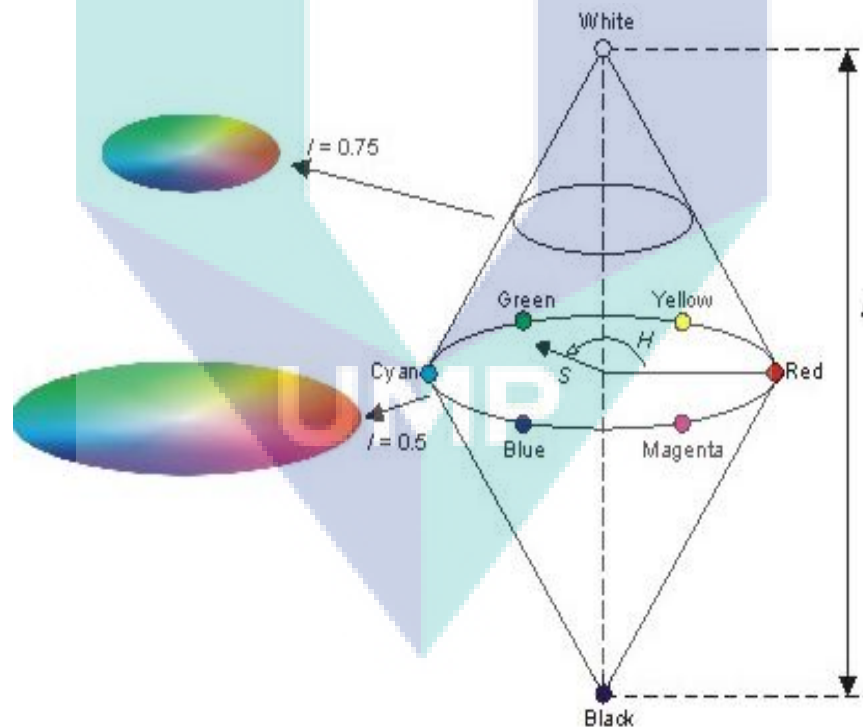
#### **2.4.2 Hue-Saturation-Intensity model**

RGB color model does not correspond to the way that people recognize colors. Therefore, the HSI and HSV color models are commonly used. The previous paper (Tan, Hong, Chang, & Shneier, 2006) (Y. He, Wang, & Zhang, 2004) had shown that HSI color spaces may offer advantages in terms of robustness against changes in illumination. The performance of RGB and HSI in identifying the road and road sign has been verified in this research (Jau, Teh, & Ng, 2008). The conclusion declares that the HSI color model is much more suited for traffic image detection because it has hue property that is not influenced by lighting condition and HSI color segmentation process shows more correct than color identification made by the RGB color segmentation.

Sun et al. (2006) conducted a color analysis of road scene images using HSI colors modeling which full color images are converted into HSI color representation. The Hue component describes the color itself in the form of an angle between (0, 360) and the range of the  $S$  component is (0, 1). The intensity range is between (0,1) and 0 means black, 1 means white, as shown in Figure 2.1. In HSI-based analysis, simple thresholds with

saturation and intensity values avoid influences of brightness on the road. HSI color model has relative simpler thresholds but problems with HSI color segmentation arise when the illumination has a non-white color such as yellow or red that is found in most street lights which can affect the camera's observation, as a result, it falls outside the required color range.

In this study Chiu & Lin (2005) and Sun et al., (2006) conducted research on color analysis of road scene images using HSI with simple threshold. Full color images are converted into HSI color representation. This paper states that HSI model could perform segmentation of lane-markings, adaptive for different lane-marking colors, indeed it can consume lower computational costs compared with using the RGB model.



**Figure 2.1:** The HSI color space

source: The HSI color space ("The HSI color space," 2011)

The feature-based detection algorithms based on color are insufficient to detect the target independently. Typically it is with the needs of other assistive technology. The apparent color of an object is not consistent in the real world with time change, thus, the image segmentation based on color requires special treatment and normalization to ensure consistency of the segmentation results. Once the illuminant color or reflectivity of road has change in an image, it can easily lead to unsatisfactory results.

### **2.4.3 Edge detection**

Many other approaches to image interpretation are based on edges, since analysis based on edge detection is insensitive to change in the overall illumination level. Edge detection highlights the contrast of image. Detecting contrast, which is different in intensity, can emphasize the boundaries of features within an image, since this is where image contrast occurs.

Lane edges in an image are the boundary between two lanes. Edge detection methods are available in the literatures such as Sobel (1978), Prewitt (1970) and Kirsch (1971). Detectors calculate the first directional derivative to ascertain the locations of the edges. The Canny detector (Canny, 1986), which is a Gaussian edge detector, is one of the most popular edge detectors in the literature and it has been widely used in many applications (Ali, 2001; Hongjian, 2002). Although the Gaussian detectors exhibit relatively better performance, they are computationally much more complex than classical derivative based edge detectors (Yuksel, 2007) and more sensitive to noise.

### **2.4.4 Ridge detection**

Different with edges-based on opposite gradient direction, a lane line is defined as points between two parallel edge segments. A ridge is the center of the line itself while smoothing has been treated. The ridge detection is to capture the major axis of symmetry of an elongated object.



**Figure 2.2** Lane markings resemble mountains; their ridges correspond to the center lines of the lane markings

López et al. (2005) explored a different low-level image descriptor, namely, the ridges. The center lines of the long bright structures appearing in grey-level road images are the lane markings. Their ridges are the longitudinal center of the painted line, as shown in Figure 2.2. Ridges stand for a measure of how much a pixel neighborhood resembles a ridge. In this work, smoothing applied by different levels depends on the image row, the objective of which is to avoid smoothing away from the line segments.

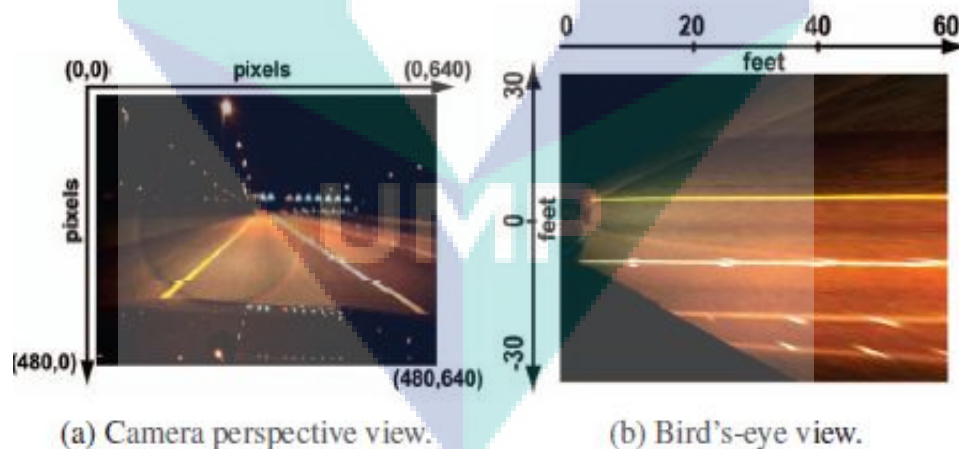
The ridges measure presents an invariance under grey-level transforms of the input image which helps lane detection in presence of shadows, however, a major disadvantage of this method is that ridges also enhance some non-spurious irregularities of the asphalt grey-level and that the Gaussian pre-smoothing kernel must be carefully tuned so as to produce a connected ridge structures.

#### 2.4.5 The Inverse Perspective Mapping

The perspective view of image somehow appears to enlarge or extend the actual space, or give the effect of distance to actual shape of road. For this reason, the image needs to go through a pre-processing stage to remedy the distortion using a transformation technique known as the inverse perspective mapping (IPM). IPM (Borkar, Hayes, & Smith et al., 2009) (Sehestedt, Kodagoda, Alempijevic et al., 2007) relied on top-view (birds-eye)

images. This method has obtained lane orientation in the world coordinates so as to give the impression of road height, width, and depth. It can convert the top-view images (camera calibration) into a two-dimensional surface. With the specific acquisition conditions (camera position, orientation, optics etc), the IPM transform can be adopted for this treating (Pomerleau, 1995).

Borkar et al. (2009) extended this work by incorporating an IPM, applying a Kalman filter to help smooth the output of the lane tracker. IPM is used to change the captured images from a camera perspective to a bird's-eye view, as shown in Figure 2.3. With this transformation, lane detection becomes a problem of detecting a pair of parallel lines that are generally separated by a given, fixed distance. In addition, this transformation enables a mapping between pixels in the image plane to world co-ordinates (feet). IPM aids in simplifying the process of finding candidate lane markers. The camera's intrinsic and extrinsic parameters are necessary to ensure an accurate transformation.



**Figure 2.3** Inverse perspective mapping transforms a camera perspective image into a bird's-eye view image.

Source: Borkar, Hayes et al. (2009)



Sehestedt et al.(2007) had presented a lane detection tracking algorithm based on weak models, which is implemented by a particle filter with the aim of tracking the multiple lane markings. The idea of the proposed method is to use an inverse perspective mapped (IPM) image to run a particle set from the bottom to the top and observe the presence of lane markings in each line. Furthermore, the filter is able to track multiple lines and to store each estimated line as a trail. This will produce a correct data association, e.g. every detected piece of lane marking are associated to one trail, which then represents the marking of one lane by a clustered particle filter. To decrease the computational effort, in this implementation 200x400 pixels resolution is adopted. As the top-view images are generated during the process, an accurate calibration of the camera, that is, a stable running environment for the system, as well as powerful calculation ability, is needed.

The IPM is used to produce a perspective effect and to force a similar distribution of information within the image plane. Road markings or objects of the same size appear smaller in the image as they move away from the camera coordinate system. The approach based on the IPM transform has the drawback that the movements of the vehicle (pitch and roll) do not allow to detect reliably if the obstacles at distances are higher than 50 meters which means it needs camera calibration in real time.

## 2.5 THE MODEL-BASED METHODS

The model-based technology is to compare portions of images to match with deformable templates or parameterized shapes. It uses parameters to represent objects that attempts to control mathematical models to fit road shape. The matching process moves the template image to all possible positions in the source image in order to find the best match between the mathematical model and the image. Matching is done on a pixel-by-pixel basis. To describe different road structure, several methods have been presented here from straight lines (Kaliyaperumal, Lakshmanan, & Kluge et al., 2001) (Schreiber et al., 2005) to (Jung & Kelber et al., 2004) (J. Wang, Gu, Zhang, & Zhang et al., 2010) and B-spline curve model (Y. Wang, Teoh, & Shen et al., 2004).



### 2.5.1 A linear-parabolic lane model

Using a linear-parabolic lane boundary model is proposed in this paper (Jung & Kelber, 2004). This model is a combination of a linear function in the near field, and a parabolic function in the far field. Locally, the road is assumed to be straight, the far field is considered as incoming curves. This model combines the robustness of the linear model with the flexibility of the parabolic model. Although the method appears to be robust under a variety of conditions, that is straight and curve road, it still cannot applied to some road structures, such as T intersection. It needs to use two models, linear and parabolic, to archive the straight line and fit curved parts of the road which is definitely led to extra calculation work.

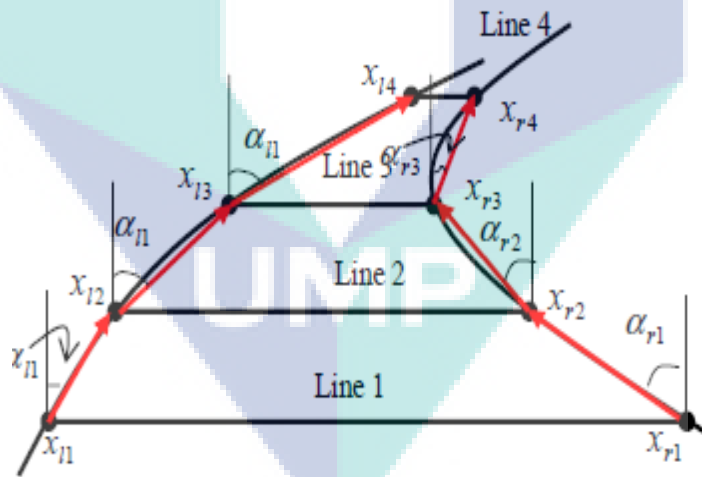
### 2.5.2 Likelihood Of Image Shape

Kreucher, Lakshmanan, and Kluge (Kreucher, Lakshmanan, & Kluge et al., 1998) designed a curve road model LOIS (Likelihood Of Image Shape), which uses a deformable template approach to detect the lane. All possible lane edges are obtained in the observed image that is described by a parametric family of shapes. A likelihood function measures how well an object shape matches. Lane detection is performed by finding the lane shape parameters that maximize likelihood of the shape parameters. Previous articles on LOIS focus solely on lane detection where the vehicle is located around the center of two lanes. This paper's contribution is using a Kalman filter to predict the future values of vehicle's location which simultaneously will consider the previous observed ones. The location is measured in terms of offset values to the right and left lane markings. If location of the vehicle on the road as determined by LOIS is within one meter, no matter whether it is the left or right lane marking or if the vehicle's path as predicted by the Kalman filter leads to a distance less than 0.8 meters from edge of the lane markings within one second, then a lane crossing warning is emitted. This approach provides a powerful means for lane crossing detection but it may not satisfy the real time purpose.

### 2.5.3 Non Uniform B-Spline

With reference to Truong, Lee, Heo, Yum, & Kim et al. (2008), Non-Uniform B-Spline (NUBS) interpolation method is used to construct left and right lane-markings of the road. Skeleton image obtained from the original one will be, selected to control the points for NUBS interpolation in constructing road boundaries. A new formulation is called vector-lane concept to extract control points. Calculate 3 angles for each lane-marking using four scan lines based on these control points obtained in the scanning lines is shown in Figure 2.4.

The advantages is that NUBS Interpolation can construct exactly any curve lane and overcome discontinuous lane markings problem. If there are very strong noises occurred, algorithm will be failed to detect the lane-marking.



**Figure 2.4** Modeling for road-lane-curvature estimation

Source: (Truong et al., 2008)

#### 2.5.4 B-Snake based lane model

With reference to Truong et al., (2008), a B-Snake based lane model is used to describe the perspective effect of parallel lines. Canny/Hough Estimation of Vanishing Points (CHEVP) provides a good initial position for the B-Snake lane model by assuming that the two sides of the road boundaries are parallel, and a set of control points are used to describe the mid-line of the road. It is able to detect a wider range of lane structures than other lane models such as straight and parabolic models. Instead of detecting two sides of lane markings, the mid-line of the lane is described by a set of control points. The problem is most of time it is not the strongest line in the images. The mid-line is commonly covered by a car in front of the camera whereby the robustness of this algorithm might be affected. Even with the enough flexibility of modeling the arbitrary shape of road, heavy commutation time of this algorithm will not be able to suit the real-time system which has a high demand of efficiency.

How to choose and maintain an appropriate shape model is a challenge subject for these techniques. Some templates are more sophisticated at the unstructured road but curve fitting usually needs more parameters to estimate that long complicated process and easy to do miss-detection. The simple straight line model is the best fitting for high speed request because of its less calculation time and better robustness on resisting noise. The complex model is more flexible and can work for more different situations but it usually costs intensive computational time and tough calculation. In the following subsection, the straight line model is discussed.

#### 2.5.5 The RANSAC (Random Sample And Consensus) algorithm

The RANSAC (Random Sample And Consensus) algorithm is a method to estimate the parameters of a chosen model starting from a set of data contaminated by outliers. The algorithm selects random subsets from the original data and uses them to obtain trial fits to the model: the best fits to the model are those which have maximum consensus with the

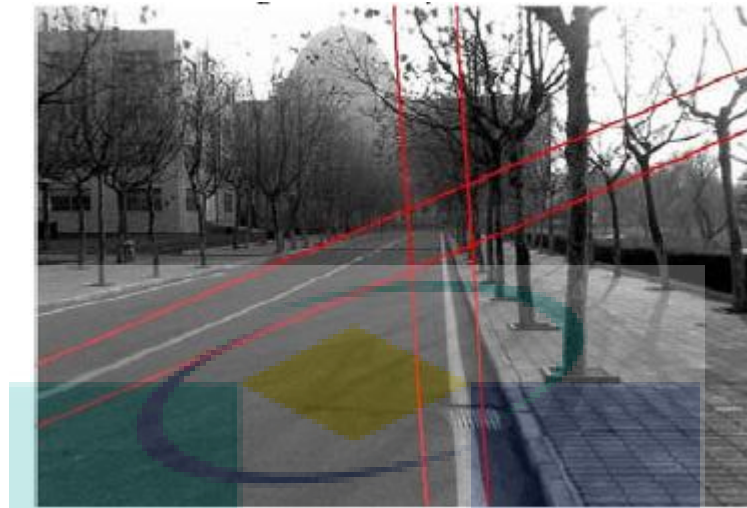
data set. The improved line detection algorithm can locate the road lane markings. Many studies of lane marking detection have been conducted, some of them using RANSAC (López et al., 2005) (Borkar et al., 2009; Kim, 2008) as line detectors and experiments help illustrate this effective method.

### 2.5.6 The Hough Transform

The Hough transform is a technique which can be used to detect lines, curves and ellipses. It is used for lane detection in most literatures for its line detection capability. The well-known Standard Hough Transform (SHT) and Progressive Probabilistic Hough Transform (PPHT) are the most efficient algorithms (Nguyen, Dai Pham, & Jeon, 2008). The algorithm consists of the initial road edge detection and the follow-up tracking of road borders. The main advantage of the Hough transform technique is that it is tolerant of gaps in feature boundary descriptions but it is relatively affected by image noise (Q. Li, Zheng, & Cheng, 2004) and computation cost still has space to improve.

Q. Li et al. (2004) had proposed a model which uses an adaptive Hough Transform. The images are first converted into grayscale using only the R and G channels of the color image. They have ignored the B channel relying on the good contrast of red and green channels with respect to the white and yellow lane markings. The grayscale image passed through with very low threshold Sobel edge detection. Afterwards, they applied a special HT which they call RHT (Randomized HT). The pixels of RHT are sampled randomly according to their gradient magnitudes. This method ensures robust and accurate detection of lane markings especially noisy images. The 3D Hough space is reduced to two dimensions for simplifying the problem and reducing the high computational cost of HT. The experiments have proven better results compared to other lane detection techniques.

According to J. Wang, Wu, Liang, & Xi, (2010), the random Hough transform (RHT) is employed to obtain the edge of the road. Get the road image at the beginning, and then pre-process the image.



**Figure 2.5:** An example of ROI  
Source: (J. Wang, Wu, et al., 2010)

Firstly calculate the ROI based on the priori, moreover, carry through the RHT on the ROI to get the follow-up frames' road edge. The follow-up frames ROI can be worked out based on the located road of the last frame. In Figure 2.5, an example of RO is an output in the video processing and the lane parts enclosed by the border are ROI. ROI calculation used in this paper can filter out most background information. However, the ROI would be slipped out of the trace area in a curve road, accordingly lead to an inaccurate detection result.

B. Yu & Jain (1997) also used Hough Transform to detect the lane boundaries. This work additionally considers the pavements at the sideways. The boundaries of the pavement can be treated as a classified continuous line. This paper has put special attention on the boundaries of pavement which can be treated as a classified continuous line. The SHT is used to find lane boundaries with a parabolic model. Road pavement types, lane structures and weather conditions have carefully been investigated. They have applied the SHT several times from a low resolution to the desired resolution images. They call this method multi-resolution SHT, and they have proven it to reduce the computational cost of SHT while preserving the accuracy. The proposed system is only tested with 34 grayscale

images of size 256 x 240. The experiments show that the system is capable of handling images of different qualities, paved and unpaved roads, marked and unmarked roads.

### **2.5.7 Other extensions to the HT**

The motivation for extending the HT is clear: keep the performance, but improve the speed (Nixon & Aguado, 2008). These approaches have included:

Kernel-based Hough transform operates on clusters of approximately collinear pixels. The voting process is weighted using oriented elliptical-Gaussian kernels to accelerate the process (Fernandes & Oliveira, 2008). An iterative randomized Hough transform (IRHT) is developed for detection of incomplete ellipses in images with strong noise (Lu & Tan, 2008). The IRHT iteratively applies the randomized Hough transform (RHT) to a region of interest in the image space. It is expected that as the RHT is iteratively applied within each newly estimated ellipse, the noise pixels are gradually excluded and better estimations obtained. This led to the development of the iterative RHT.

The Randomized HT (Xu, Oja, & Kultanen, 1990) uses a random search of the accumulator space and pyramidal techniques. One main problem with techniques which do not search the full accumulator space but a reduced space to save speed is that the wrong shape can be extracted (Yuen, Princen, Illingworth, & Kittler, 1990), a problem known as phantom shape location. These approaches can also be used (with some variations) to improve speed of performance in template matching.

## **2.6 CONCLUSION**

Overall, Model-driven approaches are powerful methods, having great research value for the analysis of road edges and markings. The Hough Transform has become the most widely used approach in model-driven approaches. There have been many approaches aimed at improving the performance of the SHT. It is admitted generally that the standard

Hough Transform has a very large computational cost associated with its voting scheme, which has prevented the performance of the system to achieve real-time request.

The desirable features of SHT technique are: first SHT must be able to resist random image noise in certain degree as random noise points usually contribute a very low level of counts in the accumulator. Second, SHT must be able to recognize aimed objects with a slight warp since its independent combination of evidence, the size and localization of peak values in the accumulator provides a measure of similarity of shape. Third, SHT must be able to tolerate the gap occlusion which is a severe problem for most shape detection techniques but the SHT tackles gracefully because the size of a parameter peak is directly proportional to the number of matching boundary and template points. The principal disadvantage is low computational speed and high storage request. The major computational cost of the algorithm is to build up the large accumulator space.

In this thesis, the standard Hough is improved by reducing search space and decreasing parameters space size based on prior-knowledge. Consequently, the computation and massive storage requirement is alleviated.

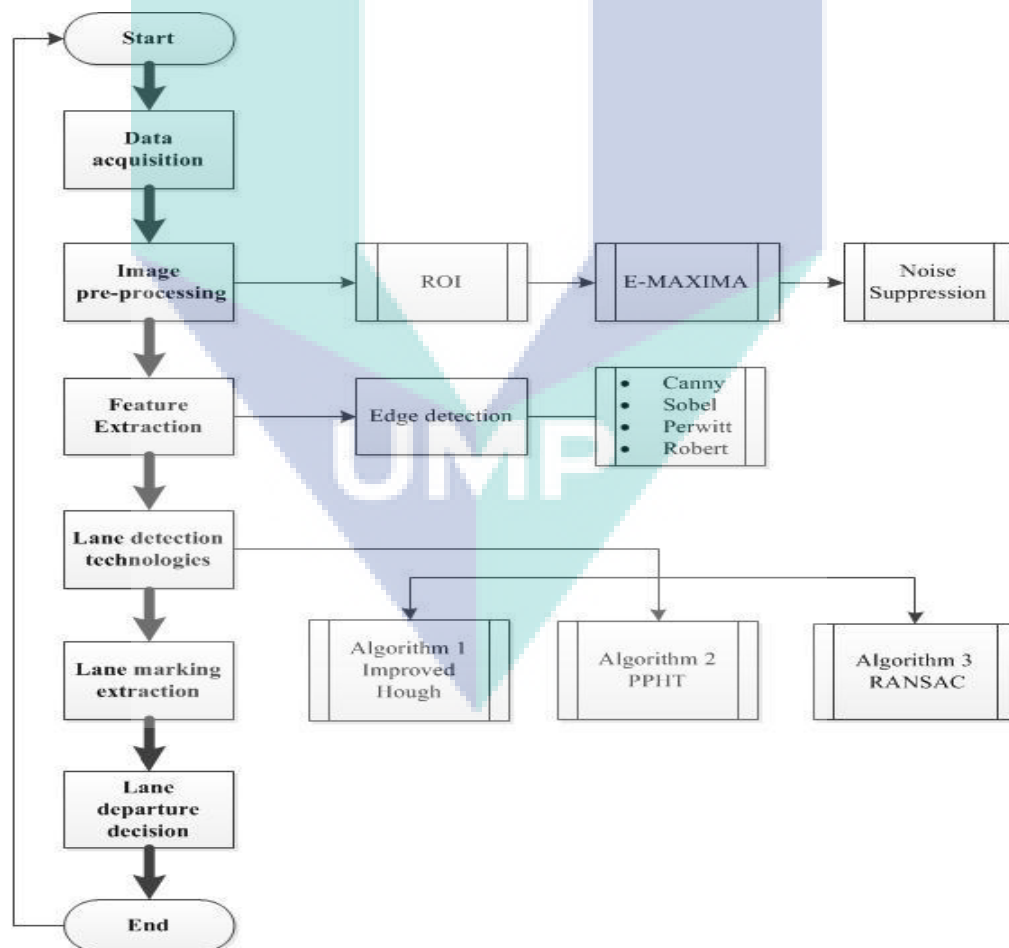
The logo of the University of Malaya (UMP) is a large, stylized 'U' shape composed of four overlapping triangles in shades of blue and green. The letters 'UMP' are written in white, bold, sans-serif font across the center of the 'U' shape.

UMP

## CHAPTER 3

### METHODOLOGY

In this chapter, a lane detection algorithm built around the Hough, Progressive Probabilistic Hough Transform and RANSAC algorithm are presented. The architecture is shown in Figure 3.1; each of the key process technologies will be explained and the basic theory will be discussed.



**Figure 3.1** The main algorithm flow-chart





**Figure 3.2** A representation of data acquisition

Source: (Internet)

After collecting and analyzing the latest research materials concerning lane detection technologies, a lane detection approach is provided here based on E-MAXIMA and improved Hough transform to extract the features of a structured road. In image pre-processing, the closer field-of-view scope is defined as a straight line model. Furthermore, the best fitting line prior-knowledge is used in lane finding process to efficiently decrease Hough space, thus enhancing the program's robustness and speed of process.

### 3.1 DATA ACQUISITION

The selection of a suitable experimental data is the first of a series of difficult problems faced in developing a lane detection system. The type and quality of data collected directly affect the subsequent processing method and outcome. 48 test videos have been selected in this experiment, each approximately 20 seconds long – in total about 24,000 frames. The video was recorded under real-time driving situation. The fixed CMOS camera was installed between the front windscreen and the rear-view mirror, as seen in Figure 3.2.

### 3.1.1 Image processing platform

In order to prove the effect and usability of proposed algorithms, video samples were collected outdoor. The image processing platform (hardware and software) are introduced here. The video was processed on a conventional PC environment. The hardware parts for running the simulation is listed in Table 3.1 with video specifications taken from the camera sensor in Table 3.2.

**Table 3.1 Hardware specifications**

Hardware	Specifications
Processor	Intel (R) Core ((TM) i3-2100 CPU @3.10 GHz
Memory	4.00 Gigabyte
Cache	3624 Megabyte
Hard Disk	300 Gigabyte
Chip Set	ATI Radeon HD 3450- Dell Optiplex

**Table 3.2 Videos specifications**

Video	Specifications
Format	Mp4/AVI
Length	20 sec
Frame-width	640
Frame-Height	480
Frame rate	28f/s

### 3.1.2 Python and MATLAB

MATLAB (Rafael, Gonzalez, Richard, & Steven L, 2004) is a programming environment ideal for technical computing that requires a large use of arrays or graphical data for analysis. The syntax of the programming code is very similar to C, and is also very

forgiving for errors made by the programmer. It is an interpreted language whose basic data element is an array that does not require dimensioning. This allows formulating the solutions to many technical computing problems, especially those involving matrix representations like image processing. The most powerful aspects of MATLAB are that many commonly used functions are already built-in to the program. For example, array-sorting algorithms, algorithms are quickly and easily implemented because of this feature. It allows MATLAB to be a very useful environment for testing out approaches to solving problems before committing them to C or Java, or other programming languages. The main intention of this application is to verify if the algorithm could be done for lane detection.

Although MATLAB is widely used in academic and research institutions, the defect of MATLAB is an closed source software, and some functions in the toolboxes are wrapped which give difficult to modify or improve MATLAB functions. There's therefore the need of a low-cost –open-source software. Python can be an alternatives tool. .

Python is an open-source object-oriented programming language, free to use. It has efficient high-level data structures and effective approach. The feasibility of our algorithms is validated by both Python and MATLAB.

### 3.1.3 Data collection

A number of criteria must be satisfied in data collection process. First, the data must contain sufficiently driving situation in different traffic density level. Second, natural scenery factors must be considered such as outdoor lighting and weather. Third, road structures are relatively abundant covering isolated roads, metro and highway. Fourth, noise, shadow, road surface cracks, driving vehicles and background objects will be examined. The details of outdoor data collections are listed in Table 3.3.

**Table 3.3 The outdoor pre-conditions of Data collection**

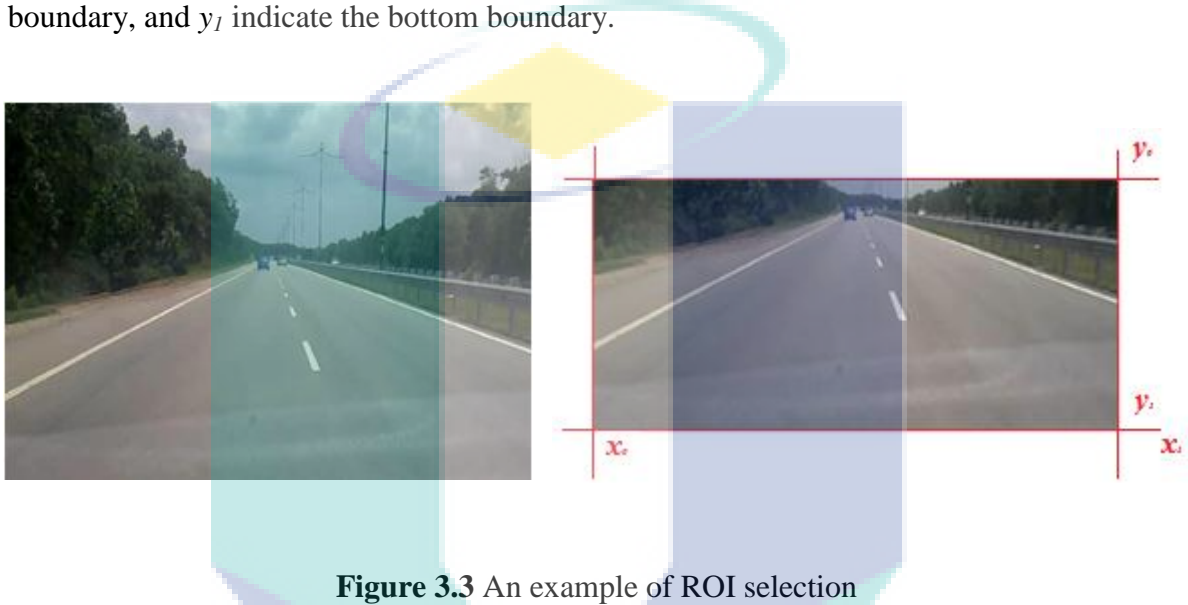
Preconditions	Description
Car speed	100km/h~120km/h
Road conditions	multiple /isolated lane
Road shapes	Straight/curve
Capture time	8am, 10 am, 12noon, 2pm, 10pm
Weather	Rainy, Sunny, Cloudy, Dusty

### 3.2 REGION OF INTEREST SELECTION

Data collection from the camera between the front windscreen and the rear-view mirror where is the source of image frames delivered to the image process unit. When the camera lens direction is parallel to the ground, the image frames taken can be divided by the foreground and background fields. Choosing an appropriate ROI will not only minimize the search area in images but also diminish the interference from extraneous objects. The farthest objects in the image frames, which are above the horizon, consist mainly of clouds, sky, hills or far distance objects, would be considered as less interested region for lane detection purpose. The greatest region of interest extends from the bottom line of the image frame to around 15 meters in front of the vehicle, where all the important objects are located, like lane markings, pedestrians and other vehicles.

In Wu (2010) and Wu, Chen, Chang, Chen, & Chung et al. (2007), dynamic ROI had been used. Researchers firstly calculated the ROI based on the initial boundary conditions on priori-frames. The subsequent frames' ROI was figured out based on the detected road lane of the previous frame, but in this sort of approaches any miss-detection happened on the beginning of the video stream will lead to a wrong ROI calculation. Therefore all sequenced detection based on that incorrect region would almost cause disastrous results when driving on the highway.

In this thesis, images are obtained by a camera with image dimensions of  $640 \times 480$  pixels, and image height  $H$ ,  $H=480$  pixels. The ROI is defined as  $2/3$  height of image with an approximation from the region below the horizon line to the boundary of the image  $H_{ROI}=H \times 2/3$ . The width of ROI is the same as the image width  $W_{ROI}=x_1-x_0$ . An example is given in Figure 3.3 where  $x_0$  and  $x_1$  is the left and right boundary of the ROI,  $y_0$  is the top boundary, and  $y_1$  indicate the bottom boundary.



**Figure 3.3** An example of ROI selection

### 3.3 FEATURE EXTRACTION BASED ON E-MAXIMA TRANSFORMATION

According to the Manual on Uniform Traffic Control Devices (MUTCD, 2009), white markings for longitudinal lines shall delineate the separation of traffic flows in the same direction. Yellow markings for longitudinal lines shall delineate the separation of traffic travelling in opposite directions.

In order to detect lane marking, the lane marking pixels must be extracted first. The strong contrast between lane markings and road surface is used to extract the lane marking edges. An extraction scheme based on the EXTENDED-MAXIMA transform (E-MAXIMA) is introduced to obtain the feature extraction result. Both minima and maxima are important morphological features as they often mark relevant image objects: minima for dark objects and maxima for bright objects so it is a perfect match with road characters

where lane markings represent maxima region in images. The E-MAXIMA is applied to decrease image noise and remove interference points as much as possible.

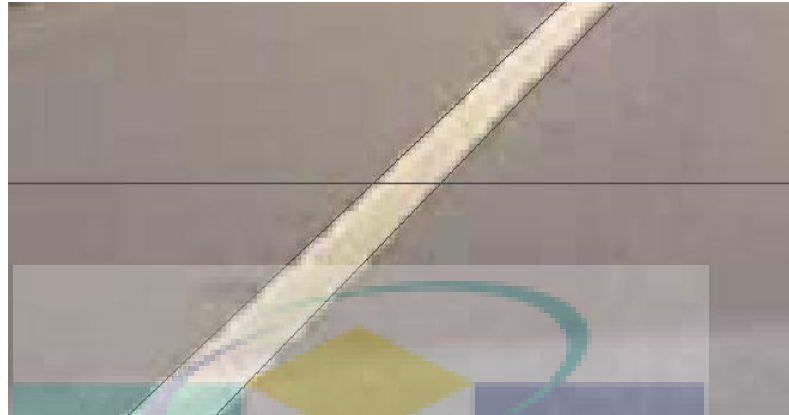
Extraneous objects can be eliminated from images by E-MAXIMA transformation; consider the intensity between road surface and lane markers has a strong image contrast. In grayscale colormap light-colors have a higher value; dark-color value is approximated to 0. E-EXTREMA transformation provides a function to filter the image extrema using a contrast criterion. More precisely, the E-MAXIMA transformation suppresses all maxima whose pixel values are lower or equal to a given threshold level will turn to the black background in binary image, on the contrary, pixel values higher than threshold level turn to bright objects (Soille, 2004). An example of road surface is shown in Figure 3.4. White lane markings represent bright objects which have distinguishable higher pixel values than the background. Most background patterns are converted to 0 (black) as they have relatively low gray value, as seen in Figure 3.5. The regional extrema is used to extract the textures by marking the extrema and non-extrema region with 0 and 1. The E-MAXIMA transformation is the regional maxima of the H-MAXIMA transformation. H-MAXIMA can be defined by the formula below:

$$HMAX_h(g) = R_P^\lambda(g - t) \quad (3.1)$$

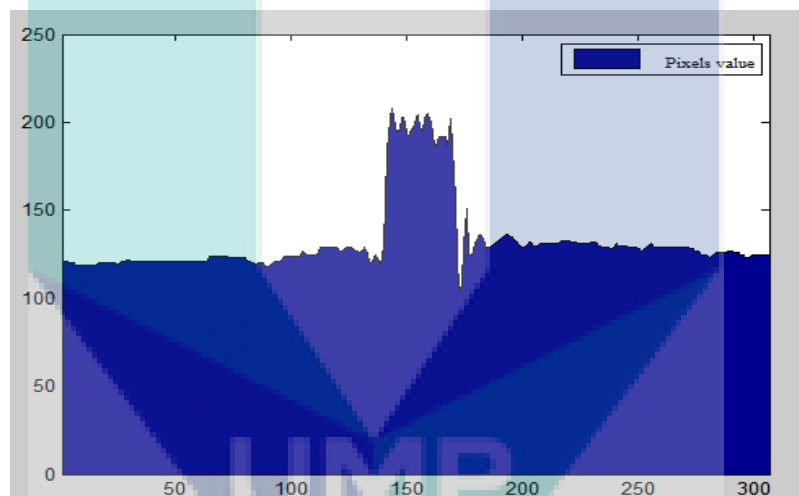
Where  $g$  expresses an intensity image; the H-MAXIMA transform  $H$  is used to suppress all maxima in the intensity image;  $t$  means threshold.  $R$  represents the reconstruction.  $R_P^\lambda$  is the morphological reconstruction by  $g$ .

The E-MAXIMA transformation can be defined in Eq(3.2). The extended minima EMIN are defined as the regional minima of the corresponding H-MINIMA transformation where E-MAXIMA is the maxima region of H-MAXIMA:

$$\begin{aligned} EMAX_h(p) &= RMAX[HMAX_h(p)], \\ EMIN_h(p) &= RMIN[HMIN_h(p)]. \end{aligned} \quad (3.2)$$



**Figure 3.4** A cross section in road-surface image



**Figure 3.5** Distribution of pixels value in a cross section.

### 3.4 EDGE DETECTION TECHNOLOGIES

Edges detection is fundamental importance in Image segmentation. Edges are strong intensity contrasts area in image, sharp, discontinuities intensity between neighboring pixels, which are usually identified between target object and background or different regions. The main contribution of detecting edges is that it filters out useless information and significantly simplifies data, while at the same time preserving the

important structural objects in the image. The result output can be transferred to a binary image where each pixel is marked as either an edge pixel or a non-edge pixel.

Majority of edge detection techniques can be categorized into two groups: search-based and zero-crossing based (Zhai, Dong, & Ma, 2008). The search-based method detects the edges by looking at the maximum and minimum in the first derivative of the image. The zero-crossing method searches for zero crossings in the second derivative of the image to find edges.

For an image  $z = f(x, y)$ , in  $x$  direction,  $y$  direction and  $\alpha$  direction, first-order directional derivative can be:

$$\begin{aligned} f_x(x, y) &= \frac{\partial f(x, y)}{\partial x} \\ f_y(x, y) &= \frac{\partial f(x, y)}{\partial y} \\ f_\alpha(x, y) &= f_x(x, y) \sin \alpha + f_y(x, y) \cos \alpha \end{aligned} \quad (3.3)$$

Image  $z$  in  $x$  direction,  $y$  direction and  $\alpha$  direction by second-order derivative can be expressed as:

$$\begin{aligned} f_{xx}(x, y) &= \frac{\partial^2 f(x, y)}{\partial x^2} \\ f_{yy}(x, y) &= \frac{\partial^2 f(x, y)}{\partial y^2} \\ f'_\alpha(x, y) &= f_{xx}(x, y) \sin^2 \alpha \\ &\quad + f_{yy}(x, y) \cos^2 \alpha + 2 f_{xy}(x, y) \sin \alpha \cos \alpha \end{aligned} \quad (3.4)$$

In digital image, the above differential operations can be replaced by the direction of different functions; first-order directional derivative is given below:



$$\Delta_x f(i, j) = f(i, j) - f(i - 1, j)$$

$$\Delta_y f(i, j) = f(i, j) - f(i, j - 1)$$

The corresponding second-order derivatives are defined by:

$$\Delta_\alpha f(i, j) = \Delta_x f(i, j) \sin \alpha - \Delta_y f(i, j) \cos \alpha$$

$$\Delta_x^2 f(i, j) = \Delta_x f(i + 1, j) - \Delta_x f(i, j)$$

$$\Delta_y^2 f(i, j) = \Delta_y f(i, j + 1) - \Delta_y f(i, j)$$

$$\Delta_{xy}^2 f(i, j) = \Delta_x f(i, j + 1) - \Delta_y f(i, j)$$

$$\Delta_{yx}^2 f(i, j) = \Delta_y f(i + 1, j) - \Delta_x f(i, j)$$

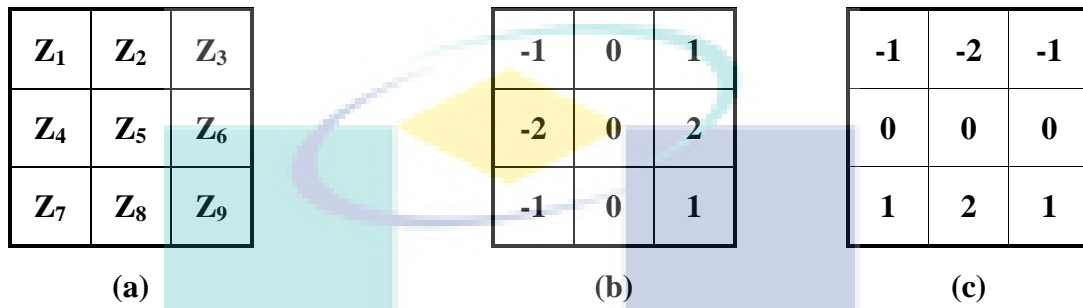
$$\Delta_\alpha^2 f(i, j) = \Delta_x^2 f(i, j) \sin^2 \alpha + 2\Delta_{xy}^2 f(i, j) \sin \alpha \cos \alpha + \Delta_y^2 f(i, j) \cos^2 \alpha \quad (3.5)$$

Therefore, the derivation of each pixel can be calculated by grayscale values within a neighborhood around the pixel. Four differential operators: Sobel, Prewitt, Roberts and Canny will be discussed in the following subsections.

### 3.4.1 Sobel operator

Sobel is a gradient operator, applies 2D spatial gradient convolution operation to an image to find its edges information. By using a pair of  $3 \times 3$  convolution kernels, Sobel calculates the approximate gradient magnitude in two directions, horizontal and vertical, giving the direction of the largest possible change from light to dark and the rate of change in that direction. These pair of kernels can be convolved separately on the source image, to get separate measurements of the gradient component in horizontal and vertical orientation,

then they are be combined to find the gradient magnitude. Defining  $Hx$  and  $Hy$  as a pair of kernels,  $Hy$  actually is  $Hx$  rotated by  $90^\circ$  as shown below:



- (a) One pixel and the surrounding pixels in image  
 (b) Convolution kernels  $Hx$   
 (c) Convolution kernels  $Hy$

If using  $I$  represents source image,  $Gx$  and  $Gy$  are two images which at each point contain the horizontal and vertical derivative approximations, the computations can be:

$$Gx = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} * I$$

$$Gy = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} * I$$

An approximate gradient magnitude is computed using:

$$\begin{aligned} Gx &= (Z_7 + 2Z_8 + Z_9) - (Z_1 + 2Z_2 + Z_3) \\ Gy &= (Z_3 + 2Z_6 + Z_9) - (Z_1 + 2Z_4 + Z_7) \end{aligned} \quad (3.6)$$

The gradient magnitude at each point can be computed by  $G_x$  and  $G_y$ :

$$G = \sqrt{G_x^2 + G_y^2} \quad (3.7)$$

Sobel edge detector is a simple and effective approach because it is using the fast convolution functions. The Sobel operator consists of two separable operations (Engel, Hadwiger, Kniss, Rezk-Salama, & Weiskopf, 2006). First, smooth perpendicular to the derivative direction then simplify central difference in the derivative direction. It is experimentally shown in this thesis that the chosen threshold is playing an important role in the edge detection process. The experiment result is discussed in the section 4.2

#### Pseudo code of edge detection using the Sobel operator

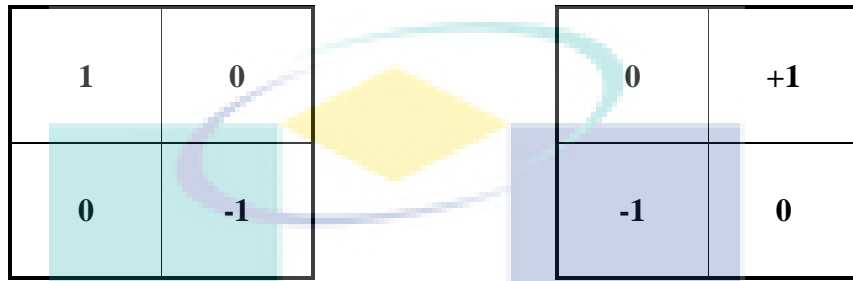
1. **Input:** A grayscale image: gray\_Img
2. **Output:** The Black and White edge image: bw\_im.
3. Initialize mask  $G_x[]$ ;
4. Initialize mask  $G_y []$ ;
5. Get image size [row col] = size(Img);
6. Apply Mask  $G_x, G_y$  to the input image.
7. **for** ( each rows) do
8.     **for**(each columns) do
9.         Using Eq (3.7)
10.         compute  $\text{sumx} = \text{sumx} + 0(\text{row} + x, \text{col} + y) * G_x(x, y)$ ;
11.         compute  $\text{sumy} = \text{sumy} + 0(\text{row} + x, \text{col} + y) * G_y(x, y)$
12.         **end**
13.         let SUM = absolute value of (sumx) + absolute value of (sumy)
14.     **end**
15.     **if** (SUM > 255)         SUM = 255;
16.     **else if** (SUM < 0)     SUM = 0;

#### 3.4.2 Roberts cross operator

Roberts operator is the simplest operator as it applies a partial differential operators to find the edge information in the image (Cunningham, Tablan, Roberts, Greenwood, &

Aswani, 2011; C. Fan & Ren, 2010). It is calculated using a set of axes rotated at 45 degrees in relation to the usual orientation of column and row.

First convolve the original image with the following pair of  $2 \times 2$  convolution kernels:



Let  $f(i,j)$  be a point in the original image,  $G(i,j)$  be a point in an image formed by convolving with the kernel. Roberts proposed equation is given by:

$$G(i,j) = |f(i,j) - f(i+1,j+1)| + |f(i+1,j) - f(i,j+1)| \quad (3.8)$$

Typically, using convolution masks, the above equation can be replaced by

$$G(i,j) = |G_x| + |G_y| \quad (3.9)$$

The kernel of Roberts Cross operator is small and contains only integers. However it suffers greatly from sensitivity to noise (Maitra, Nag, & Bandyopadhyay, 2012). According to Roberts, an edge detector should have the following properties:

- The produced edges should be well-defined.
- The background should contribute as little noise as possible.
- The intensity of edges should correspond as close as possible to what a human would perceive.

The subsection 4.2.1 Threshold Evaluation shows the effect of the Roberts operator working under three thresholds, auto, 0.1, and 0.6.

### 3.4.3 Prewitt operator

The Prewitt operator is a discrete differentiation operator, quite similar to the Sobel operator. Two  $3 \times 3$  kernels one for horizontal changes and one for vertical can be seen below:

-1	0	1
-1	0	1
-1	0	1

1	1	1
0	0	0
-1	-1	-1

$G_x$  and  $G_y$  represent two images where every point contains derivative approximations.  $I$  represent the input image,

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} * I$$

$$G_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} * I$$

The gradient magnitude computation by  $G_x$  and  $G_y$  is the same to the Sobel operator, see Eq. (3.7).

### 3.4.4 Canny Edge Detector

The Canny edge detection algorithm is known to many as the optimal edge detector (Green, 2002). The Canny edge detector processes the image to reduce noise with a Gaussian filter. After smoothing the image, the technique finds the intensity gradients of

the image, and suppresses any pixel that is not at the maximum. Below is a 5x5 Gaussian filter where  $\sigma = 1.4$

1/115

2	4	5	4	2
4	9	12	9	4
5	12	15	12	5
4	9	12	9	4
2	4	5	4	2

The process works as follows:

- (i). Smoothing the image and eliminating the noise using a filter based on the first derivative of a Gaussian filter. The larger the width of the Gaussian mask, the lower the detector's sensitivity to noise (Bansal, Saini, Bansal, & Kaur, 2012).is A 5x5 Gaussian filter is used in this implementation.
- (ii). The next step is to find the intensity gradient of the image. The orientation of edges has arbitrary angles, compared to the Sobel operator which calculates the approximate gradient magnitude in horizontal and vertical directions. The Canny detector estimates the gradient magnitude in four directions: horizontal, vertical and the two diagonals, for example .0, 45, 90 and 135 degrees.
- (iii). The magnitude of the gradient is then approximated using the formula:

$$G(i, j) = |Gx| + |Gy| \quad (3.10)$$

The formula for finding the edge direction is given by:

$$\theta = \tan^{-1} \frac{Gy}{Gx} \quad (3.11)$$

The Canny detector searches for local maxima in the gradient direction. Non-maximum suppression is used to trace the edge direction. Only local maxima should be marked as edges, while non-maxima that is not considered to be an edge will be suppressed, namely sets its pixel value to equal 0.

Edge contour in the image is not a constant value as part of the edge elements fluctuates above or below the threshold. If only one threshold is used, the marked edge might be discontinuous like a dashed line. Canny uses adaptive threshold with hysteresis; it means two thresholds high and low are involved. All points in the image which has values greater than the high-threshold are selected into edge elements, as well as any point around which have a value above than the low-threshold is also selected as edge element. Once this process is completed we have a binary image where each pixel is marked as either an edge or a non-edge pixel. The results of edge detection by Canny detector are shown in Figure 4.4.

### 3.5 LANE MODEL ANALYSIS

According to the Malaysian Public Works Department (PWD) Guide on Geometric Design of Road ("A Guide on Geometric Design of Roads, Jabatan Kerja Raya Malaysia," Arahan Teknik 8/86) the radius for the curvature of a highway has certain standards that needs to take full consideration of safety factors including speed, friction, etc. The following formula is used in determining the required minimum radius for any highway curve:

$$R = \frac{V^2}{127(e + f)} \quad (3.12)$$

Where  $R$  equals minimum radius of the curve in meters,  $V$  equals design speed, in km/h,  $e$  means maximum rate of super elevation and  $f$  represents maximum allowable side friction factor.

Table 3.4 gives the minimum radius that is to be used in road design. The guide indicates that flatter curves should always be used wherever possible, and any transition or spiral curves design should be avoided.

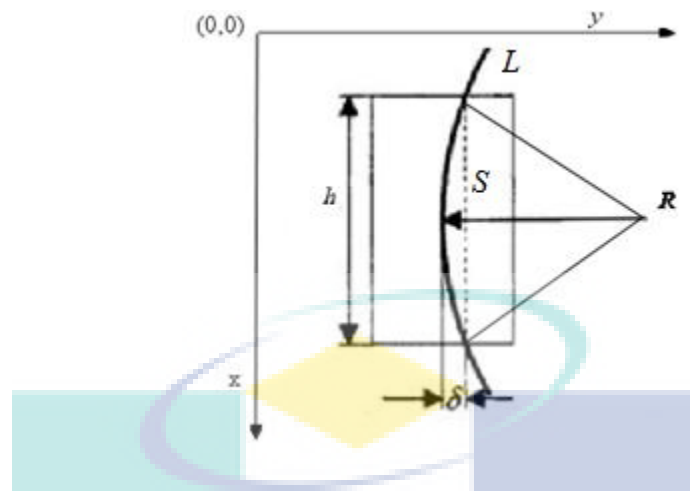
**Table 3.4** Minimum radius for road design

Design Speed (km/hr)	Minimum Radius (m)		
	e = 0.06		e = 0.10
50	100		85
60	150		125
80	280		230
100	465		375
120	710		570

Table 3.4 shows that the faster vehicle speed needs a lower road curvature. The more approximate to the straight line roadway and the value of curvature will tend to 0. The actual situation is that the highway curvature is in accordance with the guide. The curvature of a straight line is zero. The curvature should be large if radius  $R$  is small and the curvature should be small if  $R$  is large. Accordingly the curvature of a circle  $K$  is defined to be the reciprocal of the radius:  $K = \frac{1}{R}$ .

The error is the difference between using a straight line and a curve model, which is denoted in Figure 3.6, where  $L$  represents the actual curved lane marking, the straight dashed line  $S$  represents equivalent straight lane model,  $R$  is a curvature radius and  $H$  is the height of ROI.





**Figure 3.6** The difference between using a straight line and a curve model

So to calculate the error between using a straight line and a curve line, the following equation applies:

$$\delta = R - \sqrt{R^2 - (H/2)^2} \quad (3.13)$$

The minimum average radius of any highway is 85m, and by choosing an appropriate window height of  $H=2\text{m}$ ,  $R=85\text{m}$ , applying in Eq. (3.13),  $\delta=0.005\text{m}$ , this means that a simple straight lane model can fit on contour of curve road by a very minor error margin, as shown in Figure 3.7:



**Figure 3.7** An example of using straight lane model on curve road

### 3.6 LANE DETECTION TECHNIQUE

The Hough transform algorithm uses an array, called an accumulator, to detect the existence of a line. By taking a point  $(x, y)$  in the image, all straight lines pass through that point satisfy the equation below (Illingworth & Kittler, 1988; Kiryati, Eldar, & Bruckstein, 1991):

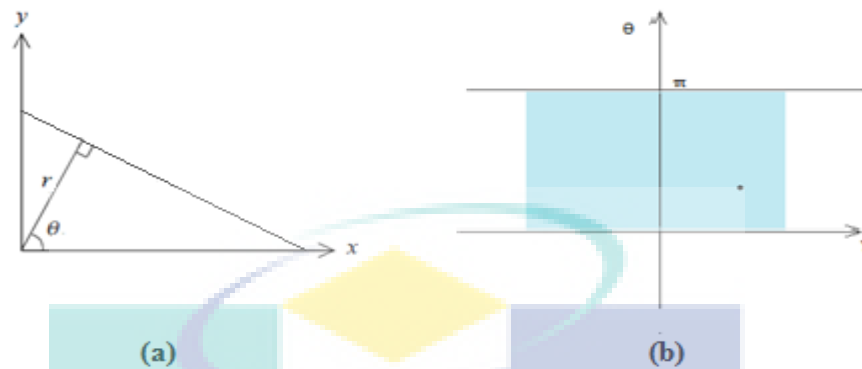
$$pX + q = y \quad (3.14)$$

The parameters  $p$  and  $q$  are the slope and the interception. Each pair of image points  $(x, y)$  through a straight line corresponds to one of the points in  $(p, q)$  parameter space. However, this equation is not able to represent vertical lines, the parameters  $(p, q)$  get up to infinite values. For this computational reason, Polar Coordinates system has projected by Hough. Therefore straight lines are presented by:

$$X \cos \theta + Y \sin \theta = r \quad (3.15)$$

Where  $r$  represents the distance between the line and the origin,  $\theta$  is the angle. If the input image is a  $A \times B$  binary array, while  $\theta \in [0, \pi]$ , the values of  $r$  and  $\theta$  are constant for each point on a line. Now for a given point in the  $(X, Y)$  plane, we can calculate the lines crossing the point in all possible angles by iterating in  $[0, \pi]$ . For the lines that go through the given point  $(X, Y)$ ,  $r$  can be determined by  $\theta$ .

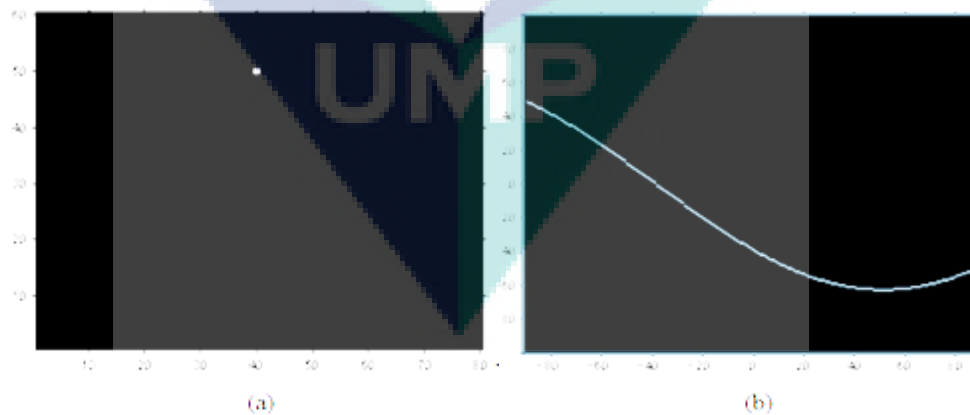
As seen in Figure 3.8 below, (a) is the representation of a line the Cartesian coordinate space using  $(r, \theta)$  and (b) show a point in the polar coordinate space correspond to a straight line in the Cartesian coordinate space.



**Figure 3.8** Mapping of one line to the Hough space

(a) A straight line in X-Y space; (b) A straight line in Polar Coordinates system

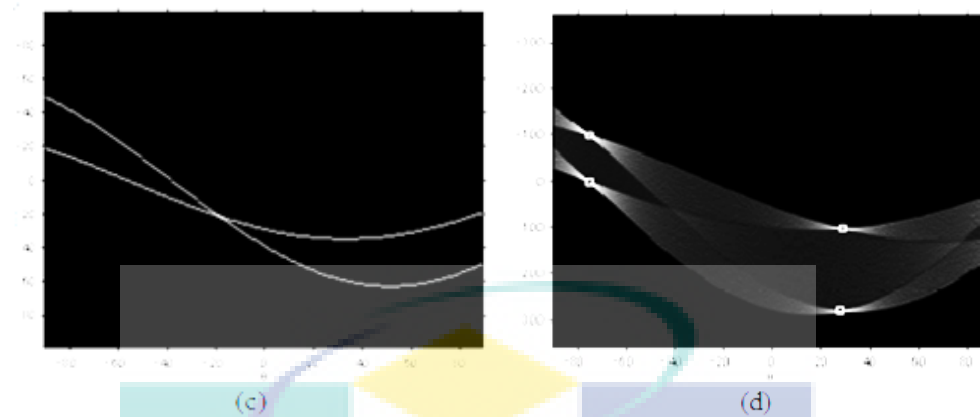
Hough transform find collinear points in an image by performing a voting procedure. It is a point to line mapping from image space to the Hough space, which maps each pixels with all possible lines that could pass through that are approximately as the sine curve. In order to illustrate how it works in detail, Figure 3.9 is given below:



**Figure 3.9.** Transformation of points to lines in the Hough space

(a) One point in image space

(b) All possible lines crossing one point plotted in the Hough space



(c) All possible lines crossing two points plotted in the Hough space  
 (d) White dots show peak value of Hough space

It can be seen from (c) and (d), that the points are mapped into concurrent lines and detecting peaks where many lines intersect. The problem about locating collinear points becomes the problem of finding peaks value in Hough space, that is, computing a global line extraction procedure has been simplified to find a local feature approach. Those peaks in the Hough space are commensurate with straight lines exist in the input image.

### 3.6.1 The algorithm1: Improved Hough based on prior-knowledge

It can be known by the theory and implements of Hough transform, the standard Hough algorithm has heavy calculation burden resulted in ineffectiveness to satisfy real time request. To reduce computation cost, this thesis use prior-knowledge, improved two aspects according to lane detection based on Hough. First, define ROI area. Hough is adopted within ROI limits, reduce image pixels which join Hough transform; second by utilizing prior knowledge to determine reasonable search range of angle  $\theta$ .

The second improvement can be explained as follows. In Hough space,  $\gamma$  is polar radius,  $\theta$  is the polar angle. Experiment analysis had proved that Hough transform could detect the polar angles which fluctuated narrowly in a continuous stream flow. The search range of  $\theta$  will be restricted based on the detected  $\theta$  in the Hough transformation of the previous image frame. Thus the polar angle search scope is defined in the region near  $\theta$ ,

for example the search scope can be  $(\theta_{min} + T_\theta, \theta_{max} - T_\theta)$  where  $T_\theta = 0.261$  is the threshold. In this way it reduces the computation time and has the effect of reducing the number of useless voting process, by this way the visibility of the real lane information in the image is enhanced.

The detailed description of each step is presented below:

- (i). Initialized frames in first 5 frames were obtained, using as the prior knowledge to determine a reasonable search range of  $(r, \theta)$ . A discrete parameter space between its maximum and minimum values is established.
- (ii). Create and initialize a two dimensional accumulator array  $H(r, \theta)$  with zeros.
- (iii). Scan in ROI of the image and collect edge points where the edge pixels have been assigned a value of 1. Sample points set will be selected in turn until all elements are processed; traversing the  $\theta$  value and calculating the value of  $r$ .
- (iv). Thus get the values of parameters  $(r, \theta)$ ;  $H(r, \theta) = H(r, \theta) + 1$ . The array of  $H(r, \theta)$  is accumulated by 1.
- (v). By repeating the procedures described above, the array  $H(r, \theta)$  has been built up. Abandon  $\theta$  values out of range  $(\theta_{min} + T_\theta, \theta_{max} - T_\theta)$  then find the maximum voting values of the accumulator  $(r, \theta)$ ; determine the T strongest straight line.

### Pseudo code of Improved Hough

1. **Input:** A Black and White edge image: bw\_im.
2. **Output:** Accumulator  $H(r, \theta)$ .
3. initialize ROI to image size
4. initialize accumulator  $H(r, \theta)$  with zeros
5.  $\theta_{min}=0, \theta_{max}=180$ . Value of  $\theta$  in the range 0-180.
6. Add  $(r, \theta)$ .into a vector.
7. **If** vector size >10, calculate the valid\_lines vector  $(\theta_1, \dots, \theta_n)$ .
8. **end if**

```

9.   for image from 1 to end columns
10.      for image from 1 to end rows
11.         if input image pixel (column,row)==white
12.            for  $\theta = \theta_{min}$  to  $\theta_{max}$ , do
13.               computer  $r$  using Eq (3.15)
14.               If  $r > 1$  and  $r < \sqrt{\text{rows}^2 + \text{columns}^2}$  then
15.                   $H(r, \theta) = H(r, \theta) + 1$ 
16.                  If  $\theta$  value in  $(\theta_{min} - T_\theta, \theta_{max} + T_\theta)$ .
17.                     Add  $(r, \theta)$  to valid line vector
18.                  end if
19.               end if
20.            end for
21.         end if
22.      end if
23.   end for
24.   Find peak value in the accumulator  $H$ , to find lines.
25.   for every  $\theta$ , do
26.      for every  $r$ , do
27.         If  $H(r, \theta) > T$  (threshold), then  $(r, \theta)$  is a line
28.         end if
29.      end for
30.   end for
31.

```

### 3.6.2 The algorithm2: Progressive Probabilistic Hough Transform(PPHT)

The PPHT is a variation of SHT (Standard Hough transform), the key difference between SHT and PPHT is that SHT counts the maximum voting values of the accumulator after all possible points in image space mapping to the Hough space; The idea of PPHT is to transform randomly selected pixels in the edge image into the accumulator. When a bin in the accumulator corresponds to a particular infinite line which has got a certain number of votes, the edge image is searched along that line to see if one or more finite lines are present. It is designed to speed up the SHT and minimizing the computation requirements.

In this section, PPHT is implemented using the HT approach to process the entire vision data in order to detect the lines

The procedure of PPHT is listed below:

- (i). An accumulator  $H(r, \theta)$  is initialized to 0 and all the candidate boundary points in the image space of ROI are added into an input set  $P$ .
- (ii). Randomly select a point  $P(i, j)$  from the input set  $P$ , and then vote into the accumulator  $H((r, \theta))$ .
- (iii). Remove those points which have been selected from  $P$ , until the input list  $P$  is null, then the loop will stop.
- (iv). Find the maximum value of the accumulator  $H((r, \theta))$ . If the accumulator is lower than threshold  $T_h$ , go back to step (ii).
- (v). If the maximum value in accumulator is higher than threshold  $T_h$ , then accordingly search  $P$  along a straight line specified by  $H(r_{peak}, \theta_{peak})$  to find the longest segment, then add into  $P_{max}$ .
- (vi). Remove from  $P$  all points in  $P_{max}$ , clear  $H$ .
- (vii). If  $P_{max}$  is longer than a given minimum length, add  $P_{max}$  to the output list. Otherwise, go to step (i).

#### Pseudo code of PPHT

1. **Input** :points data set  $P$ .
2.  $T_h$  threshold of enough aligned points.
3.  $T_{min}$  , given minimum length
4. **Output** lines clusters
5. Initialize accumulator  $H(r, \theta)$
6. **While**  $p$  is not null **then**

### Pseudo code of PPHT (continued)

```

7. Randomly select a points  $P(i,j)$  from  $p$ 
8.   If threshold  $T_h$ , reached then
9.       Vote into the accumulator  $H((r,\theta))$ .
10.      Find the peak value of the accumulator  $H(r,\theta)$ 
11.      search segment in  $P$  along  $H(r_{peak},\theta_{peak})$ 
12.      add the longest segment into  $P_{max}$ .
13.      If  $P_{max}$  is longer than  $T_{min}$  , then
14.          add  $P_{max}$  to output
15.      end if
16.
17.      remove  $P_{max}$  from  $P$ .
18.      remove  $P(i,j)$  from  $P$ 

19.   end if
20. end while

```

### 3.6.3 The algorithm 3: Random Sample Consensus(RANSAC)

In this thesis, RANSAC is used as an alternative technology to implement lane marking detection, for comparison with Hough. RANSAC is a widely used iterative sampling algorithm. An advantage of this algorithm is its robust fitting ability to model parameters. It simply iterates two steps: generating a hypothesis from random samples and verifying it to the data. Different with conventional sampling techniques that use as much of the data as possible to obtain an initial solution and then proceed to prune outliers, RANSAC uses the smallest set possible and proceeds to enlarge this set with consistent data points (Mahbub, Imtiaz, & Rahman Ahad, 2011). RANSAC estimates model parameters by the minimum number subset from observed data points. Assuming that data points consist of inliers and outlier, inliers can be fitted to model approximately and outliers represent those points do not fit the model. Procedure start from small number of data is based on the presupposition that it belongs to inliers. This will check the number of data



points if they fit this model. Too few points marked as inliers, this model is dropped, then generates or refines another model. Keep repeating a certain number of times, until we get a model that optimally fits data points.

- (i). Randomly enrolled a sample of data points from  $S$  and instantiate the model from this subset.
- (ii). Determine the set of data points  $S_i$  which is within a distance threshold  $t$  of the model. The set  $S_i$  is the consensus set of samples and defines the inliers of  $S$ .
- (iii). If the subset of  $S_i$  is greater than threshold  $T$ , re-estimate the model using all the points in  $S_i$  and terminate.
- (iv). If the size of  $S_i$  is less than  $T$ , select a new subset and repeat the above.
- (v). After  $N$  trials the largest consensus set  $S_i$  is selected, and the model is re-estimated using all the points in the subset  $S_i$ .

We can choose  $n$  sample data points to produce a model, assuming  $P$  is the probability of picking up only inliers from the input data. Thus, probability  $1-P$  means that at least one sample outlier is selected. Usually,  $P$  is set to a high value so that the probability  $1-P$  can be smaller, e.g.,  $P=0.99$ .  $P$  is defined as

$$P = S^n \quad (3.16)$$

Where,  $S^n$  is the probability that all  $n$  points are inliers.  $S$  is the probability of choosing an inlier from a single point;  $n$  is the points selected to generate a model.

The probability of all  $n$  points is outliers which may therefore be rewritten as

$$1 - p = (1 - S^n)^i \quad (3.17)$$

Where,  $i$  is the number of iterations. After logarithm conversion, the equation is given below:

$$i = \frac{\log(1 - p)}{\log(1 - s^n)} \quad (3.18)$$

#### Pseudo code of **RANSAC**

```

1.  input:
2.    image
3.    S - a sample of s data points
4.    Si - the subset data points , estimated a model fits to data
5.    T - threshold, the minimum number points fit the model
6.    n - is the number of iterations
7.
8.  output:
9.    Largest_consensus_set – Best fitting data points estimated by model.
10.  n := 0
11.  largest consensus := 0
12.  while i < n
13.    Si := model parameters fitted to inliers
14.    if point fits model with an error smaller than t
15.      add point to consensus_set
16.      this_error = a measure of how well points fitted
17.    end if
18.    if Si > model parameters fitted to all points in Largest_consensus_set
19.      Largest_consensus_set := Si
20.      error := this_error
21.    end if
22.    iterations add 1
23.  end while
24.  return Largest_consensus_set, error

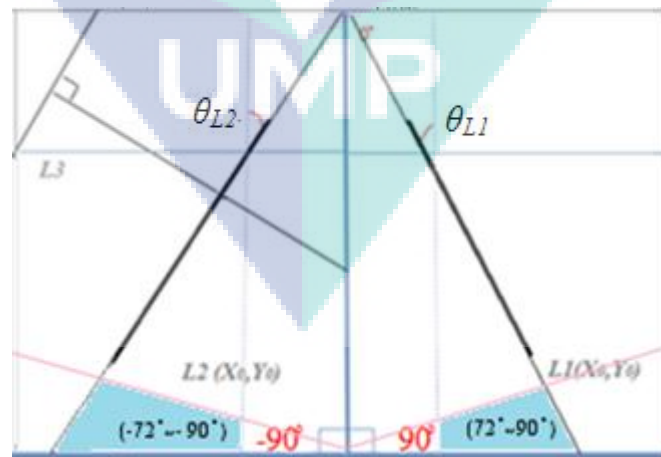
```

### 3.7 LANE MARKING EXTRACTION

After using linear detection technologies to extract lines, the next step is to determine the lane marking position and abandon false lines information, which is the subject of this section. This study proposes a method to measure the distance between

bottom points of a detected line with  $\frac{1}{2}$  width of image, and computing the angle of lines. When the result satisfied the following rubrics, the detected line will be considered as a lane marking, otherwise the line will be considered as an invalid detection to be abandoned.

- (i). Suppose lines  $(L_1, L_2 \dots L_n)$  are detected, if  $72^\circ < \theta_{L1} < 0^\circ$  &&  $L_1(x_0) < \frac{1}{2} \times I_w$ ,  $(X_0, Y_0)$  represent the point near to the baseline of image,  $I_w$  is the width of image,  $L_1$  will be determined as a left side of lane marking while the car is in normal driving status.
- (ii). If  $-72^\circ > \theta_{L2} > 0^\circ$  &  $L_2(x_0) > \frac{1}{2} \times I_w$ ,  $L_2$  will be determined as a right side of lane marking.
- (iii). If  $|L_2(x_0) - L_1(x_0)| \leq 10$  &&  $|\theta_{L2} - \theta_{L1}| \leq 10$ ,  $L_2$  will be considered as invalid edges of lane marking, then continue to locate the next line which can be chosen in order from  $(L_2 \dots L_n)$ .
- (iv). If line  $L_3$  exists and the distance from  $L_3$  to the central point of image is  $r_3$  which satisfied the condition, where  $I_w$  is the width of image,  $I_h$  is the height of image,  $L_3$  position will not be considered as the current lane markers. It will be dismissed, as shown in Figure 3.10.



**Figure 3.10 A profiled lane marking illustration**

### Pseudo code of Lane marking extraction

```

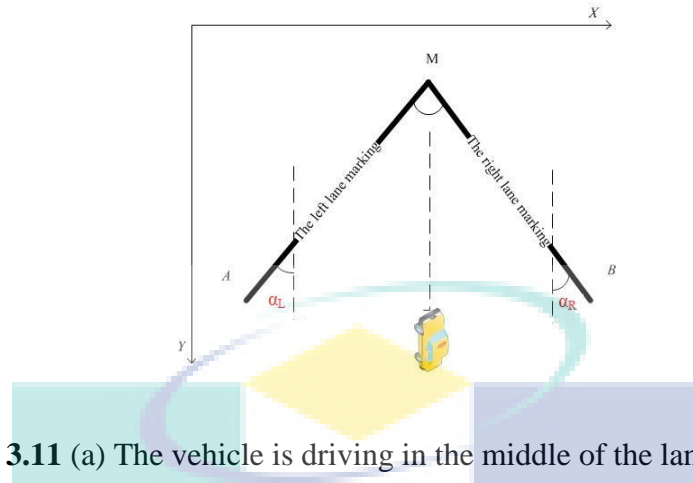
1.  input: Lines vector ( $L_1, L_2 \dots L_n$ );
2.  Output: lane mark
3.  Set  $I_w$  = width of image
4.  Get( $L_1, L_2 \dots L_n$ ) from Hough line detection
5.  While Lines ( $L_1, L_2 \dots L_n$ ) is not null then
6.    Get  $L_i$ , a one line form vectors.
7.    if the line  $L_i$  position  $r$  is out of tolerance
8.      dismissed  $L_i$ 
9.    end if
10.   Get  $L_i (X_0, Y_0)$  near to baseline of image
11.   if  $72^\circ < \theta_{L_i} < 0^\circ$  &&  $L_i (x_0) < 1/2 \times I_w$ 
12.      $L_i$  is in car left side
13.   end if
14.   if  $-72^\circ > \theta_{L_i} > 0^\circ$  &&  $L_i (x_0) > 1/2 \times I_w$ 
15.      $L_i$  is in car right side
16.   end if
17.   end while
18.   If  $|L_{right}(x_0) - L_{left}(x_0)| \leq 10$  &&  $|\theta_{L_{left}} - \theta_{L_{right}}| \leq 10$ 
19.     the next line will be load form Lines ( $L_1, L_2 \dots L_n$ ).
20.   end if

```

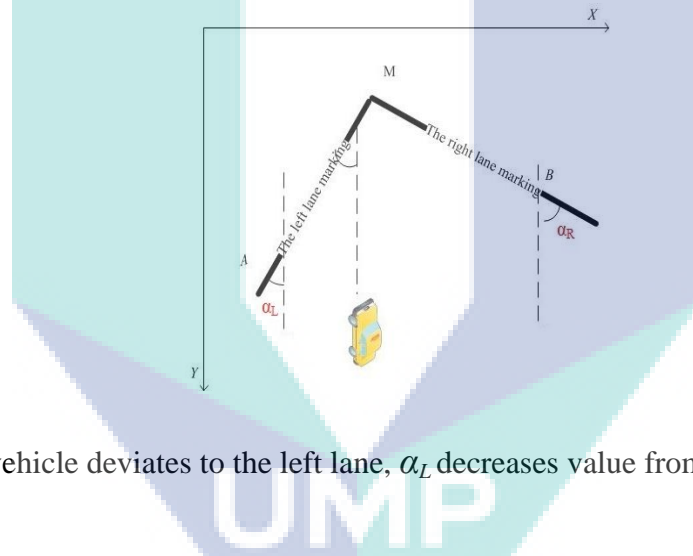
### 3.8 LANE DEPARTURE DECISION

The slopes variations of lane marking in image coordinate system differ a lot while vehicles are in a normal and abnormal driving state. The departure information of the host vehicle can be judged by the yaw angle of the right and left lane markings.

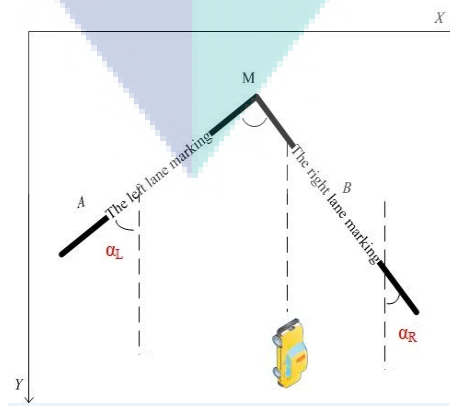
Figure 3.11(a) shows the vehicle driving at the center of the lane, consider that the camera is attached on the rearview mirror, its optical axis is carefully aligned with the central axis of the vehicle so the symmetrical orientations of left and right lane markings can be recorded.



**Figure 3.11** (a) The vehicle is driving in the middle of the lane,  $|\alpha_R| = |\alpha_L|$



(b) The vehicle deviates to the left lane,  $\alpha_L$  decreases value from  $\alpha_L > 0$  to  $\alpha_L < 0$ .



(c) The vehicle deviates to the right lane,  $\alpha_R$  decreases value from  $\alpha_R < 0$  to  $\alpha_R > 0$ .

$MA$  is defined as the left lane markings;  $MB$  the right lane markings. The slope of  $MA$  is  $K_L$ ; the slope of  $MB$  is  $K_R$ ; the angle between  $MA$  and axis  $Y$  is  $\alpha_L$ . The angle between  $MB$  and axis  $Y$  is  $\alpha_R$ .  $\alpha_L$  and  $\alpha_R$ . Study the following formula:

$$\begin{aligned}\alpha_L &= \tan^{-1} K_L \\ \alpha_R &= \tan^{-1} K_R\end{aligned}\tag{3.19}$$

(a) When the vehicle is running in the middle of the lane,  $|\alpha_R| = |\alpha_L|$ , the slope of lane markings will be  $K_L = -K_R$  and  $K_L > 0$ ,  $K_R < 0$ . When the vehicle gradually deviates to the left lane, the algebraic value of  $K_L$  and  $\alpha_L$  will decrease,  $\alpha_L$  decreases values from the range of positive numbers decrease to the range of negative, from  $\alpha_L > 0$  to  $\alpha_L < 0$  as shown in Figure 3.11(b).  $\alpha_R$  and  $K_R$  keep increasing while the vehicle gradually deviates to the right lane;  $\alpha_R$  increase values from the range of negative to positive numbers, value from  $\alpha_R < 0$  to  $\alpha_R > 0$  as shown in Figure 3.11(c). No matter whether the car deviates to the right or to the left lane,  $|\alpha_R + \alpha_L| > 0$ . With the growth of vehicle deviation,  $|\alpha_R + \alpha_L|$  continually expands to a bigger value. Hence a yaw angle  $\beta$  can be obtained by  $\beta = \alpha_R + \alpha_L$ . The lane deviation can be effectively inspected by tracing  $\beta$  value in every frame captured by the monitoring camera. In general, a car cannot strictly move in the middle line, a certain level of sway is normal. By setting a threshold value  $T$  for  $\beta$ , only when  $\beta > T$ , a lane departure alarm will be triggered.

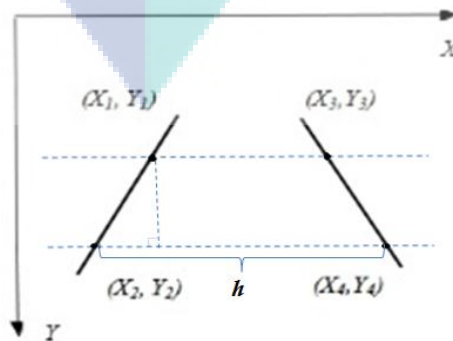
The experiment results are shown in Section 4.5 comparison of the SHT, improved Hough and PPHT.

### 3.9 THE ERROR-CATCHING MECHANISM

This subsection discusses how to capture and measure detection errors while the massive image data are processed. The lane marking position is analyzed in order to discover runtime errors to enhance the system's stability. It is worth to note that, while not trivial, that the whole process is automatic and not driven by the user.

This mechanism is specifically designed to measure accuracy and validity of lane detection result. The function of auto self-estimating procedure is to trace the detection result and give a system error in log files when miss detection occurs. Results of each frame can be recorded and estimated automatically via this procedure. The outcomes for each frame are validated according to the following rules:

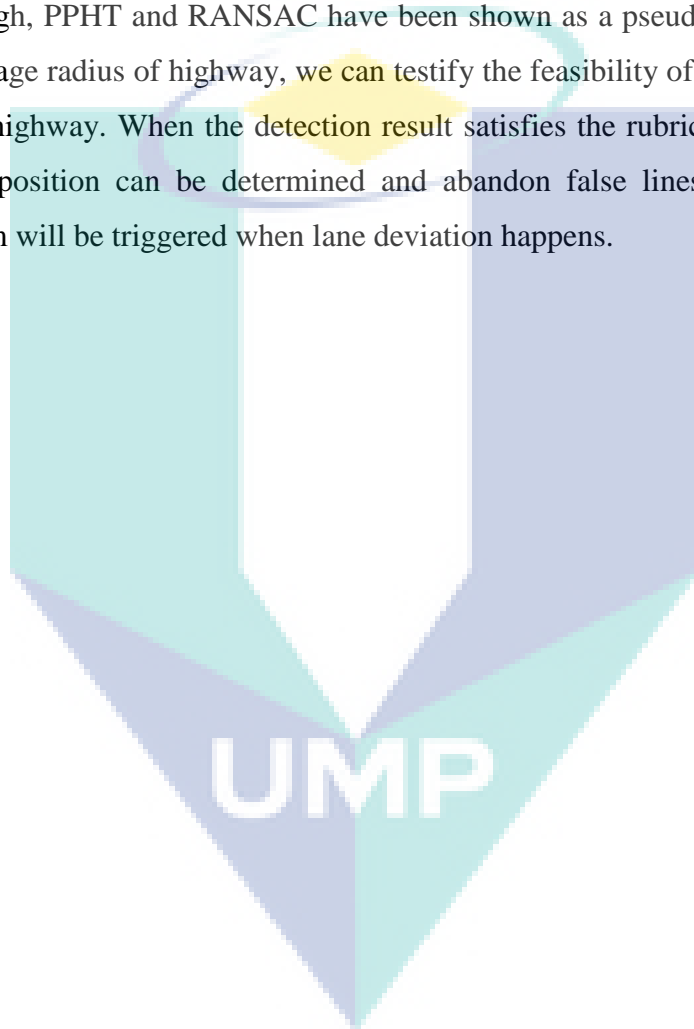
- (i). Distance between right and left lane markings has certain value in a continual image sequence. The automating tests will identify any sudden inconsistent distance change. If it is more than the threshold it would be listed as a false negative or positive result during lane detection process.
- (ii). In Figure 3.12, two points are selected from lane  $x_2$  and  $x_4$ , where the distance value is described by a length  $h = |x_4 - x_2|$ . The difference between current frame and the previous one will continuous monitoring or log errors by program whenever the distance values occur abnormal change.
- (iii). For the host vehicle it moves with constant orientation in each test section. Slope of lane markers are on normal stable distributions. The host vehicle Drifts out in the video sequence should be a gradual procedure. Slope can be described as  $S = (x_2 - x_1) / (y_2 - y_1)$ . Any inordinate slope calculated from the result and have met with the previous condition that if the distance change between the two lane markings is more than threshold, then it will be listed as a system error.



**Figure 3.12** Distance  $h$  between right and left lane marking

### 3.10 CONCLUSION

The architecture and flow chart of lane detection techniques are presented in this chapter. E-MAXIMA transform mainly contributes to decrease image noise, remove interference and extract lane markings at the primary stage. Details of algorithms, the improved Hough, PPHT and RANSAC have been shown as a pseudo code. By computing minimum average radius of highway, we can testify the feasibility of using the straight lane model on the highway. When the detection result satisfies the rubrics of the angle or line, lane marking position can be determined and abandon false lines information. A lane departure alarm will be triggered when lane deviation happens.





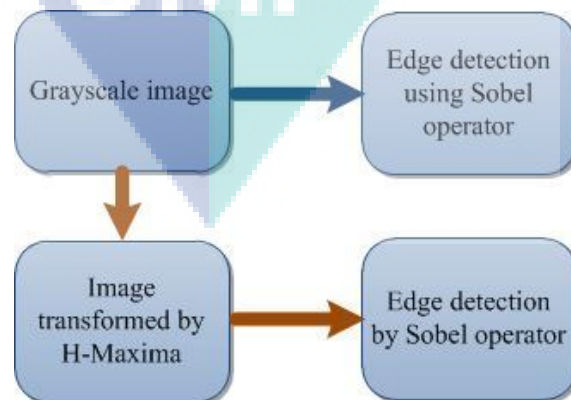
## CHAPTER 4

### RESULT AND DISCUSSION

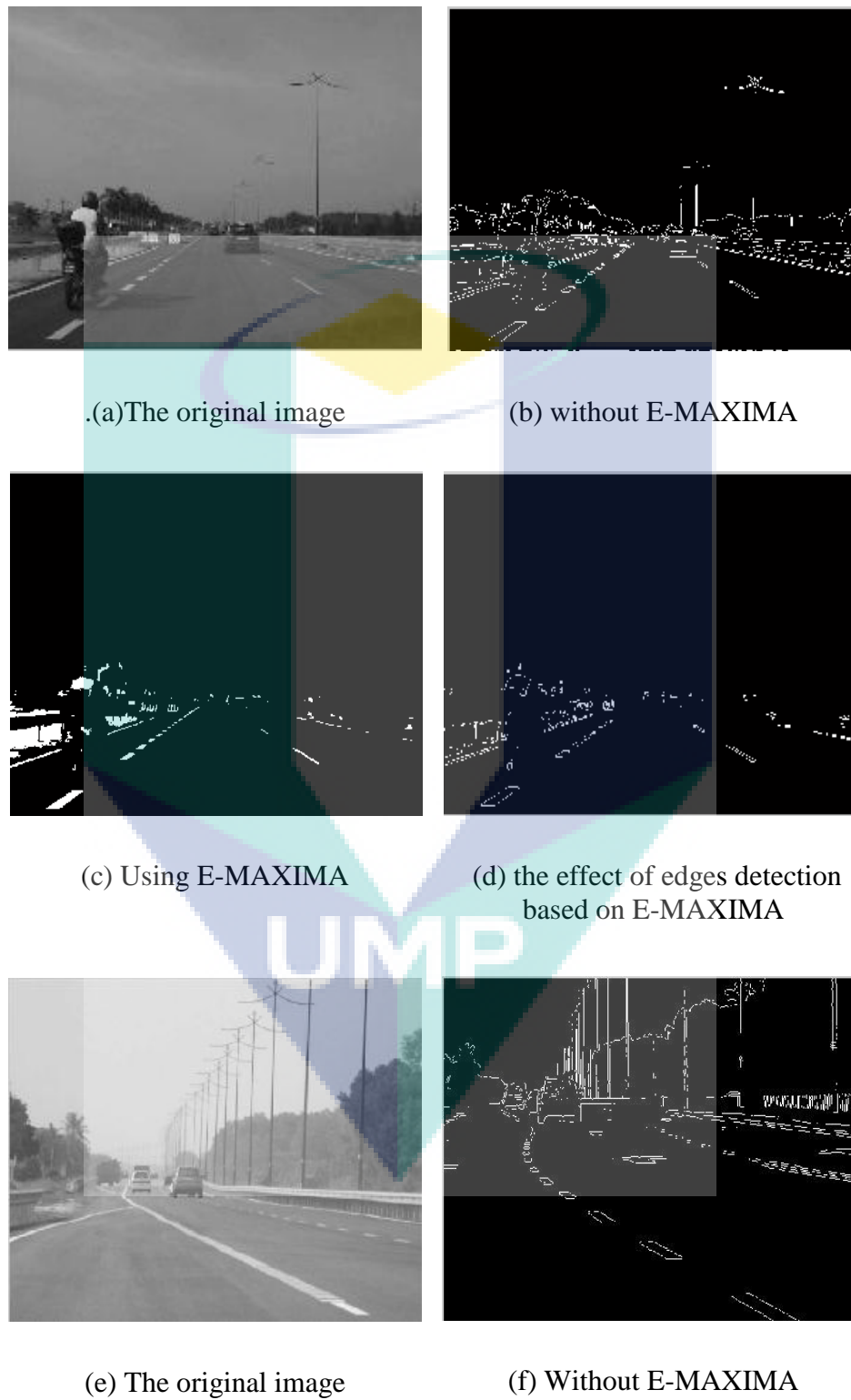
In this chapter, discussion will be focusing on the result of several experiments and comparisons using improved Hough, PPHT and RANSAC algorithms described in the previous chapter. The proposed experiments were implemented to compare the lane detection algorithm under different environmental conditions. Both visualized and numeric evaluations had been performed. Evaluation and analysis are categorized in three viewpoints: accuracy, computing time, and robustness on lane detection scenarios. The trace result and detection errors measurement were explained in section 3.9.

#### 4.1 THE EXTENDED-MAXIMA EFFECT

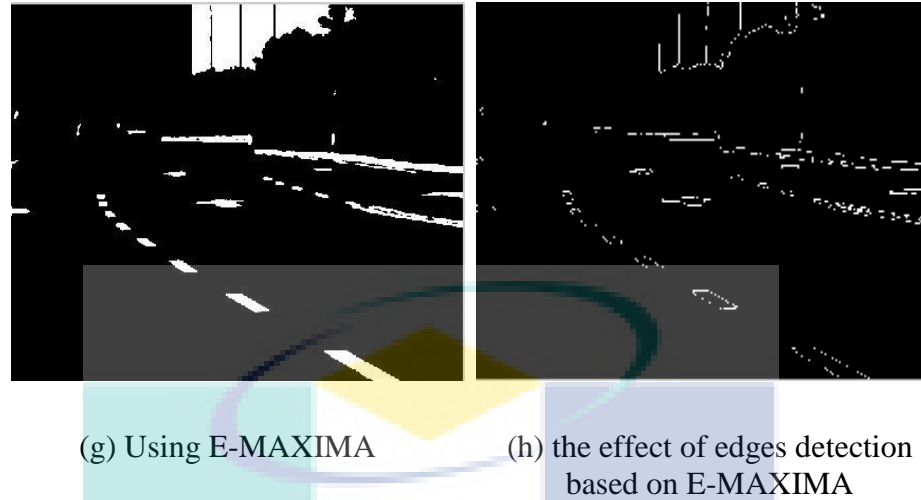
The effect of EXTENDED-MAXIMA (E-MAXIMA) presented in the experiment below shows the differences between using edge detection technologies with E-MAXIMA and without using it. The overall process is outlined in Figure 4.1:



**Figure 4.1** Process of E-MAXIMA effect evaluation



**Figure 4.2** Examples of edge detection after E-MAXIMA



**Figure 4.2 Continued**

Figure 4.2 (b) and (f) show the edge detection result without using E-MAXIMA. (c) and (g) display the output from E-MAXIMA. (d) and (h) are edge detection which follow the E-MAXIMA transformation. Judge from visualization, (d) and (h) have reached a pretty ideal binarization result which significantly reduce noise interfering. Most of the backgrounds include trees and street lamp poles are converted to 0 (black) for their relatively low gray value. In (e), the street lamp poles were not filter out, on the contrary, those poles are converted to a straight line with high pixel values. It will create obstacles for subsequent feature extraction. In (g), around 70% of street lamp poles are filtered out. With the existence of ROI, the interference can be eliminated to a relative low level.

E-MAXIMA can perform well in low level illumination conditions which can strongly enhance the contrast on the image. Two examples are given in Figure 4.3 where Figure 4.3 (b) and (d) can be found in areas of lane markings with sharp edges and high contrast will be established by using the Improved Hough Transformation.



**Figure 4.3** Applying E-MAXIMA transformation on low level of illumination

There are many advantages for image binarization using E-MAXIMA. Usually gray images can mix high relativity information of lane markings with low relativity information, for instance, the background and the noise. E-MAXIMA can directly extract the target object from the multi-valued digital images, only the high gray scale of the road markings can be preserved. It is definitely an effective way to identify the road borders which can enhance the system's robustness and minimize the negative influence from noise.

## 4.2 EXTRACTION OF LANE EDGES

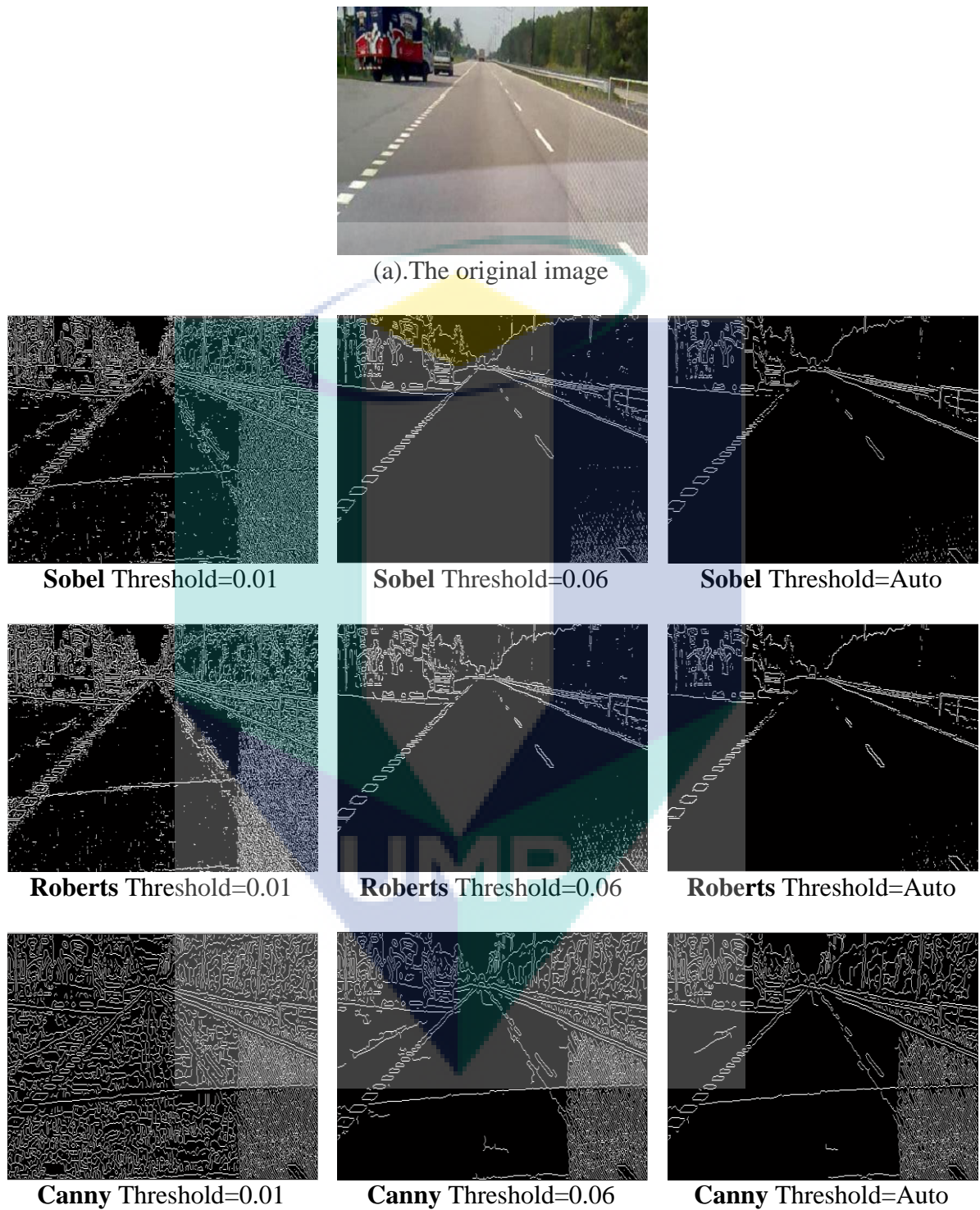
The experiment is aimed at aiding the selection of an appropriate operator that is capable of lane markings extraction. For the visual scene in driving, the performance measure for the edge detection operators is to see how well the edge operators match with the visual perception of lane boundaries. An assessment criteria is listed below:

- (i). The operator is able to find accurate edge points, less time-consuming.
- (ii). Noise fluctuations can be prevented from the edge markings.
- (iii). The stability, different orientations, length or width edges can be detected.

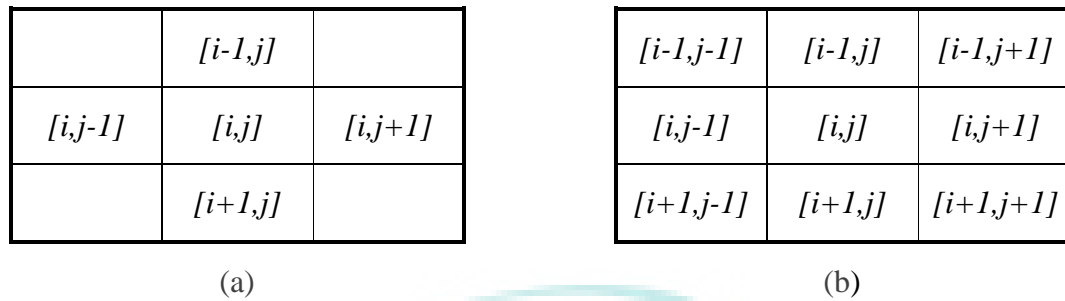
### 4.2.1 Threshold evaluation

The range of the edge detection threshold values is -1020 ~1020 for all grayscale range 0~255. To identify which threshold is the best suited to a lane detection task, the experiment indicates that using threshold from 0.01 to 0.06, the width of lane marking edges can reach 1~ 2 pixels. By applying this level of thresholds value, we can archive the purpose of thinning edges and obtain the edges enhancement effect. Hence the threshold values 0.06, 0.01 and auto have been presented here to explore the effect on the estimated threshold value. In Figure 4.4 if the threshold value is 0.01, then more edge details are detected. When compared to a higher threshold value of 0.06, less edge details can be acquired. The outcome denotes that the auto threshold can give a suitable detection result not only to find the precise regions but also the insensitive noise. The lane marking edges are relatively salient on the road surface which can be recognized by all four operators with threshold in three levels.

The Roberts cross operator in Figure 4.4 indicates the edge positioning is not accurate in the Threshold=0.01 and 0.06. However, specifying the threshold value as auto can comparatively get a better result. The Canny detector is competitive to weak edge but slightly sensitive, more easily to get influenced by extraneous boundaries other than lane markings once the edge gradient is determined.



**Figure 4.4** The edge map resulted by different threshold values



**Figure 4.5** The neighborhood structure

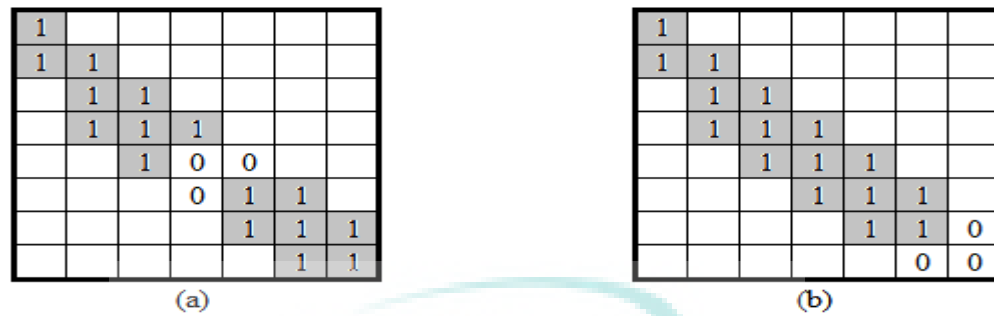
(a) 4-connection components (b) 8-connection components

#### 4.2.2 Edge connectivity analysis.

This experiment is used to analyze edge connectivity in edge map (Bowyer, Chang, & Flynn, 2006; Pande, Bhadouria, & Ghoshal, 2012). An edge map is a binary image which has only two values 0 and 1. The objective of extracting the edge information is to find out the accurate number of edge points whose pixel value is 1. We define the number of total edge points as (A), number of 4-connected components as (B) and number of 8-connected components as (C). The ratios (C/A and C/B) reflect the level of edges connectivity. 4-connected components means having continuous density in 4 neighbors of the pixels; the definition of 8-connected components is similar as shown in Figure 4.5. For edge detection case, the smaller value of C/A and C/B shows that the edge connectivity is better.

Assuming that all intensity value is 1 in an edge map, namely, the 4-connected components C is 1 and 8-connected components B is 1. The ratios of C/A and C/B have obtained the minimum value, in this case, the level of edge connectivity is the best.





**Figure 4.6** An example of linear edge markings

Figure 4.6 represents two-edge marking. Their numbers of edge point are equal to one other. In (a),  $C/A=2$ ,  $C/B=2$ . In (b),  $C/A=1$ ,  $C/B=1$ . Where (b) shows a smaller value which means the edge continuity of (b) is better. The same comparative result can be obtained based on visuals; edge of (b) is longer therefore becomes a good edge. Likewise, any other edge shape's connectivity can be calculated by  $C/A$  or  $C/B$ .

Edge detectors are assessed by statistical measures, as seen in Table 4.1. The statistical result is based on the edge map of Figure 4.4.

**Table 4.1** The connected components in edge map

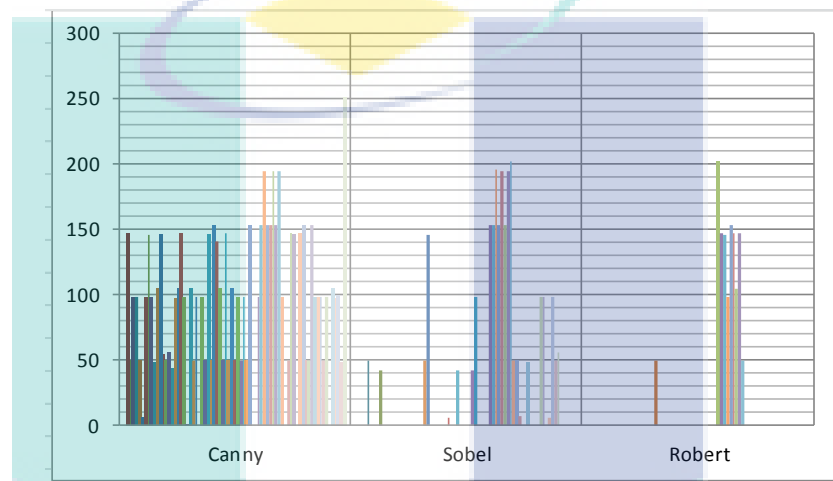
	Sobel	Roberts	Prewitt	Canny
$A(\times 10^4)$	0.892	0.7684	0.925	1.6042
B	4237	3595	4025	6390
C	424	453	444	664
$C/A$	0.0475	0.0590	0.0488	0.0414
$C/B$	0.10	0.1260	0.1103	0.1039

The result in Table 4.1 clearly show that for driving scenes, the Sobel, Prewitt, and Roberts and Canny get approximately  $C/A$  and  $C/B$  value, therefore the edges of lane



marking can be obtained with a relatively high continuity by all four types of edge detectors. The Canny operator reports the slightly high continuous edge pixels. Non-significant difference exists between the Prewitt and the Sobel. Further analysis about robustness and computation speed will proceed in the next subsection.

#### 4.2.3 Anti-noise capability analysis



**Figure 4.7** A histogram compares the edge maps resulted from operators.

In Figure 4.4 the snipping off the road region is used to measure the noise level of the edge map. Figure 4.7 shows the pixel values distribution and intensity changes after detection of edges of lane marking. Since Robert and Sobel are less sensitive, they will abandon trivial edge information and receive concentrated edge information from the road surface image. On the contrary, Canny has the ability to distinguish tiny edge information but sensitive to small objects leading to interference of recognizing lane markings.

#### 4.2.4 Comparison of edge detection efficiency

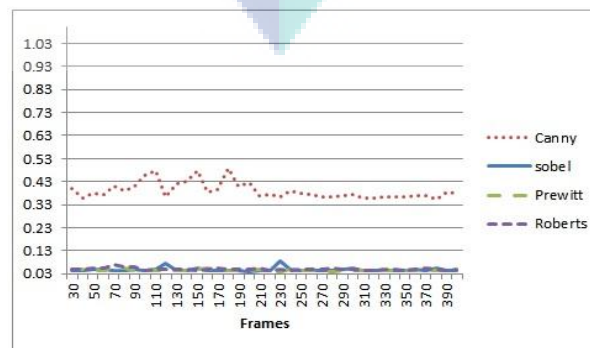
Table 4.2 shows a comparison of the four classical derivative operators on average running time to extract the edge features, where the same video data are processed by the same Hough detection algorithm but based on different edge operators to compare the

correct rates at day and night times. Correct rates of edge operators are statistical numbers how many lines can be detected by Hough. It also reflected the edge operators' effect on lane detection result. The comparative result on the performance of edge detection denotes that the Sobel operator sustains slightly high accuracy at day time and it is more suitable for lane detection purpose, hence, it suits well to filter out useless noise and inexpensive computation time.

**Table 4.2** Comparison of classical derivative operators on average running time

Edge Detectors	Sobel		Prewitt		Roberts		Canny	
	Day	night	Day	night	Day	night	Day	Night
<b>Computation time</b>	0.055	0.118	0.057	0.092	0.063	0.098	0.409	0.2498
<b>Correct rates</b>	98.6%	87.2%	98.2%	86.9%	98.17%	86.3%	97.4%	96.7%
<b>Missing rates</b>	0.02%	8.4%	0.02%	8.8%	0.02%	8.82%	0.02%	1.8%
<b>Result</b>	Optimal in day						Optimal in night	

The average computation time is shown in Figure 4.8 where Canny takes more time than the other three operators because of its complicated operation cost. Sobel, Prewitt and Roberts are relatively simple; therefore, their processing times are very close to each other.



**Figure 4.8** Average computation time using the four type of edge operators

In conclusion, the Canny operator performed better on night time under weak illumination but its computation process is more complex and more time-consuming comparing with the other 3 operators. A relatively high continuity of lane edge can be obtained by canny operator but other three types of edge detection operator also are able to satisfy the purpose of lane edge detection. Sobel operator performed better in the computation cost and it also provides a satisfying effect in noisy images. Furthermore, its extracted edges continued, localized properly and had less occurrence of miss-detect points for driving scenario.

### 4.3 LINE DETECTION TECHNIQUE

The implementation details about SHT, improved Hough and PPHT will be introduced. The experiments are designed to assess the relative merit for different algorithms; some comparative work will be done in the succeeding section.

#### 4.3.1 Behavior of the line fitting

The highest values in the Hough accumulator will correspond to strongest lines in the image. The result of the line locating and fitting is analyzed in this experiment. Each lane in a specific image is represented by an individual  $(r, \theta)$  pair. The  $\theta$  value is discretized as  $(0, 1, 2, 3 \dots 178, 179)$ . The line detection is implemented by searching peaks from the Hough accumulator. The accumulator matrix contains the number of times for each value of  $(r, \theta)$  s. Table 4.3 shows the top 8 voting results. It means they contain the greatest number of points in line.

**Table 4.3** The maximum voting result from the Hough accumulator

$\theta \backslash r$	0.6807	-0.4712	-0.2443	-1.0472	0.4712	0.2443	1.1868	-1.0123
173	157	5	12	0	3	21	27	8
248	1	71	0	0	0	0	4	0

Table 4.3: Continued

1	24	21	<b>52</b>	36	21	52	2	31
-1	0	18	30	<b>40</b>	0	0	0	35
21	21	13	30	0	<b>31</b>	31	13	0
396	8	0	6	0	11	<b>26</b>	0	0
10	12	24	24	0	13	19	<b>21</b>	21
-160	0	0	0	0	0	0	0	<b>10</b>

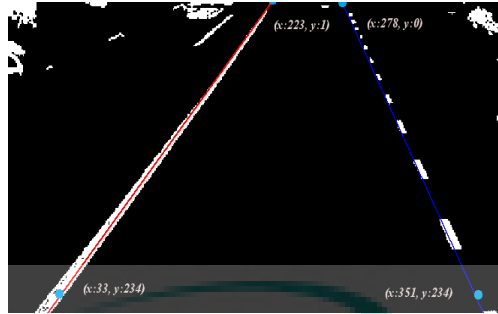
The selected highest values for  $(r, \theta)$  that satisfy the formulation:

$$X \cos \theta + Y \sin \theta = r$$

$(x, y)$  obtained from the formulation is used to describe the lines in the image.  $(x_0, y_0)$  and  $(x_1, y_1)$  are the calculated points of intersection between the reference image boundary lines and the line specified by a  $(r, \theta)$  pair, as listed in Table 4.4.

Table 4.4 The result of detected lines

	Degree	$(X_0, Y_0)$	$(X_1, Y_1)$
Line1	39°	(223,0)	(33,234)
Line2	-27°	(278,0)	(398,234)
Line3	-14°	(1,0)	(59,234)
Line4	-60°	(0,1)	(403,234)
Line5	27°	(24,0)	(0,46)
Line6	14°	(408,0)	(350,234)
Line7	68°	(27,0)	(0,11)
Line8	-58°	(0,189)	(73,234)



**Figure 4.9** The plotted pixels in the target image

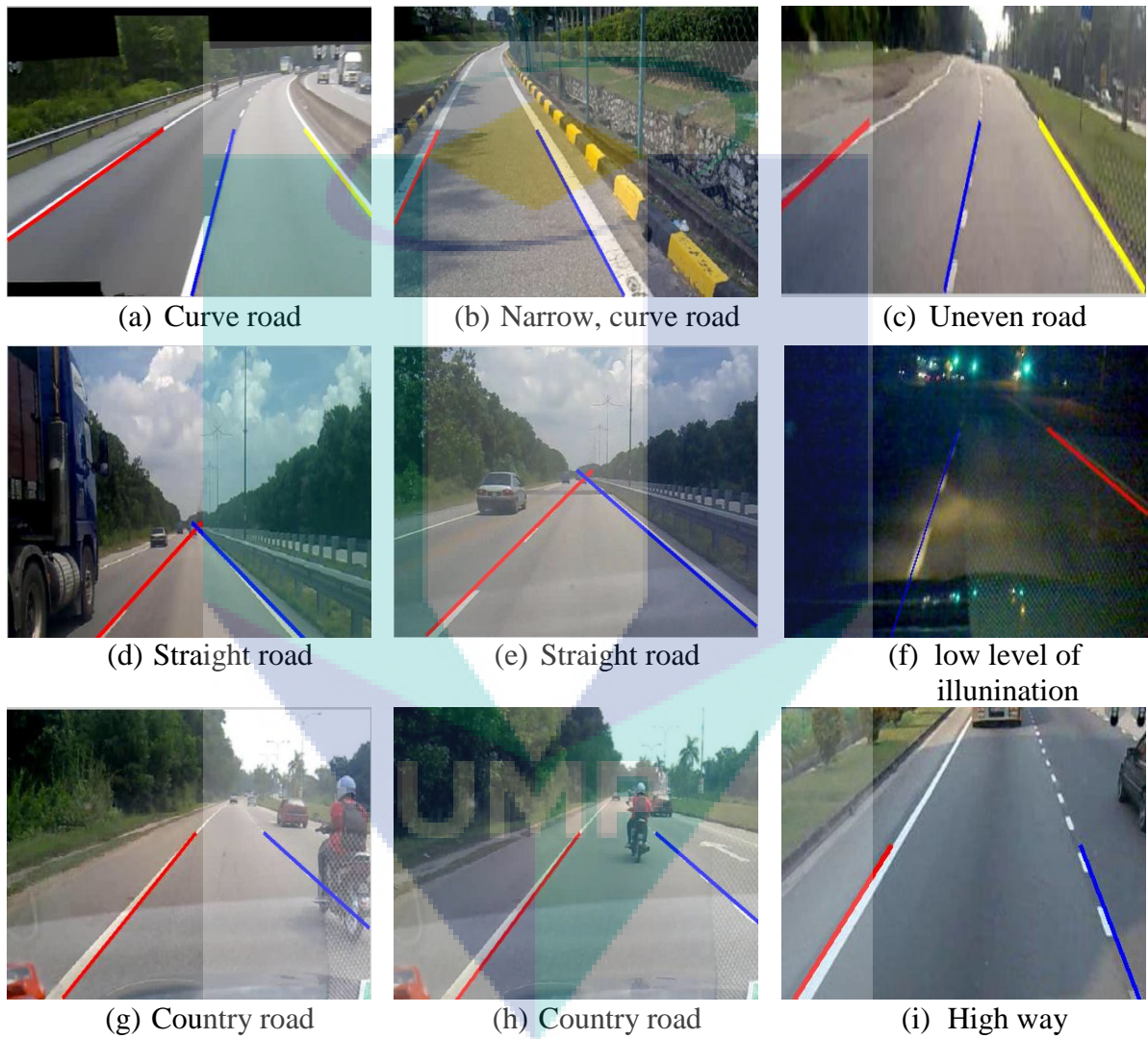
According to the inclination degree of lines, if the deviation angle  $\beta$  is less than  $25^\circ$  (refer to Section 3.8 for the deviation angle's measurement), the top of two lines with highest voting values from the Hough matrix will be selected to be plotted. If the deviation angle  $\beta$  is more than  $25^\circ$ , the top of three lines, right, middle and left lane markings will be plotted. In this case, the deviation angle  $\beta$  is less than  $25^\circ$ , so only Lines 1 and 2 (left and right lane markings) are selected and marked in the target image, an example give in Figure 4.9.

The lane detection algorithm presented has been tested on both straight and curved lanes. Figure 4.10 shows the different types of roads, shapes with variety shadows and noises. In (b) and (g), the lane marking is interrupted by shadow or motorcycle; those disconnected lane markings still can be recognized in the result page. In (f), the experiment denotes that the algorithm can recognize lane markings at a low level of illumination and achieve fairly good performance.

#### 4.4 EXPERIMENT OF LANE DEPARTURE DECISION

On the basis of experimental data, under normal and abnormal diving circumstance, the inclined angles of left and right lane markings have significant differences in view of the image coordination system. Abnormal deviation from the lane can be judged by the lane marking angles. To test deviation result from this experiment, the lane departure parameters

in Figure 4.11 illustrates the lane detection result occurred in the process of typical traffic experiment with deviation scenarios. The relationship between the lane feature's parameters and the deviation angle calculation results is shown in Table 4.5.

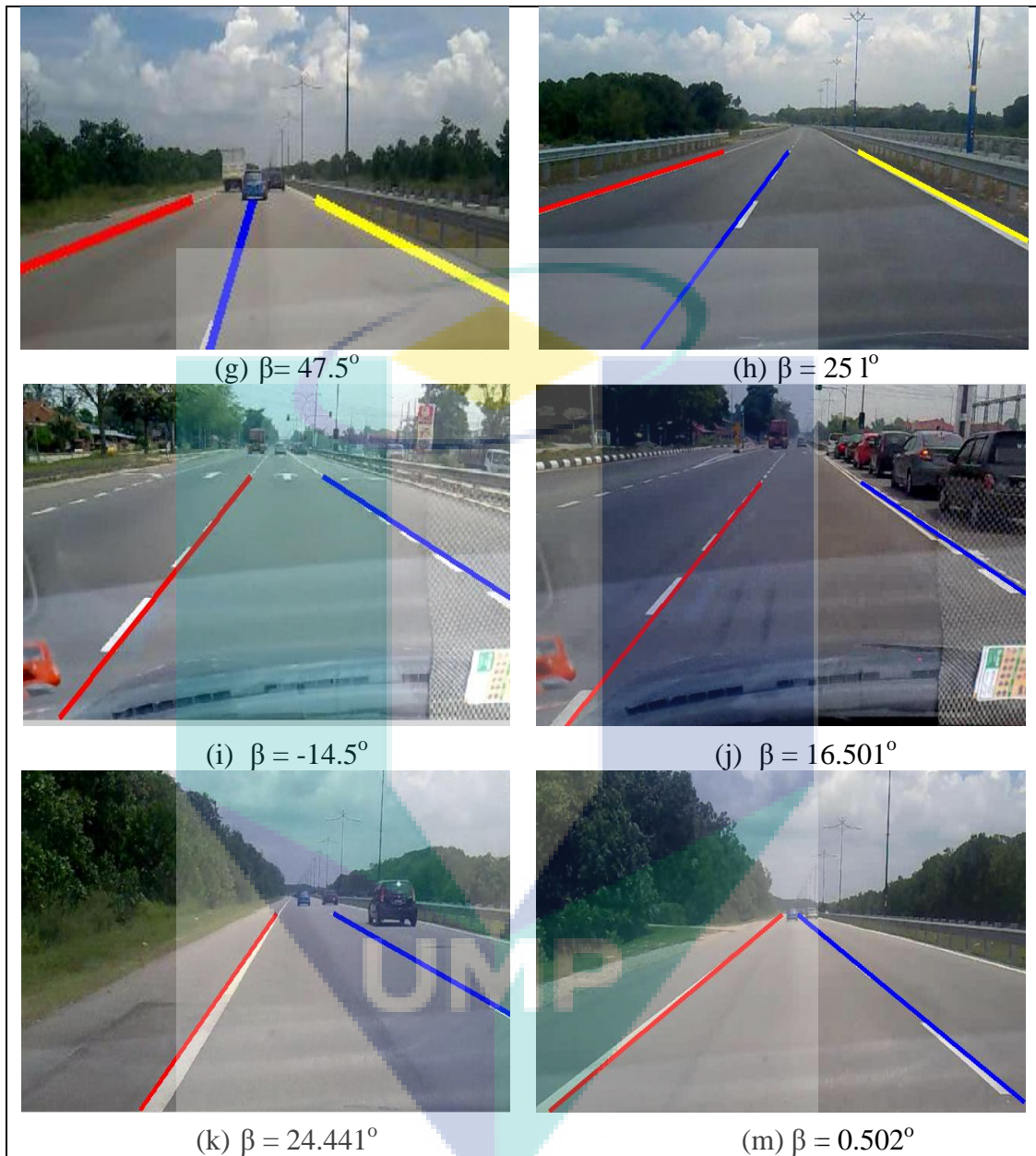


**Figure 4.10** The detection results on different roads





**Figure 4.11 Lane departure parameters extraction**



**Figure 4.11** Continued

Table 4.5 demonstrates when the vehicles are going to depart from the current lane to the next lane. The deviation angle apparently is bigger than the normal driving angle. The car deviation angle is set by value of  $\beta$ . When  $\beta < -25^\circ$  or  $\beta > 25^\circ$ , the car will divert to



another lane. The results are consistent with the actual driving situation which is a proof of the validity of this analysis.

Under inconsistent illumination and a diversity of road conditions, the accuracy and efficiency of developed algorithm have been improved greatly.

**Table 4.5** The relationship between lane markings and deviation angles

Frame	Left markings		Mid markings		right markings		Deviation Angle $\beta$	Driving Status
	$\alpha_L$	$k_L$	$\alpha_M$	$k_M$	$\alpha_R$	$k_R$		
(a)	65.99	1.1519	-34.497	-0.6021	-76	-1.3265	31.493	Yaw to right
(b)	69	1.2043	-20.99	-0.3665	-75.498	-1.3177	48.01	Yaw to right
(c)	72.5	1.2654	-24	-0.4189	-75	-1.309	48.5	Yaw to right
(d)	68.5	1.1956	-7.998	-0.1396	-73.499	-1.2828	60.502	Yaw to right
(e)	73.997	1.2915	31	0.5411	-69.499	-1.213	-38.499	Yaw to left
(f)	75	1.309	39.998	0.6981	-66.49	-1.1606	-26.492	Yaw to left
(g)	72.5	1.2654	24.998	0.4363	-69.499	-1.213	-44.501	Yaw to left
(h)	77.498	1.3526	42.5	0.7418	-67.5	-1.1781	-25	Yaw to left
(i)	-61	-1.0647			46.5	0.8116	-14.5	Within the lane
(j)	-60.96	-1.064	-	-	45.498	0.7941	15.461	Within the lane
(k)	-65.5	-1.1432	-	-	41	0.7156	-24.5	Within the lane
(m)	-55.5	-0.9687	-	-	58.99	1.0297	3.49	Within the lane

#### 4.5 COMPARISON OF THE SHT, IMPROVED HOUGH AND PPHT.

This experiment aims to compare the performance of SHT, Improved Hough and PPHT. The number of voting operations used to process real lane marking images is listed in Table 4.6. Voting and computation time are used to measure the algorithms performance.

**Table 4.6** The computational efficiency for lane marking Images

#	Voting times					Computation time		
	SHT $\times 10^5$	PPHT $\times 10^5$	Reduced %	Improved Hough $\times 10^5$	Reduced %	SHT	PPHT	Improved Hough
1	6.140	1.652	73.099%	1.263	79.43%	0.0273	0.0136	0.0216
2	5.159	1.308	74.637%	1.388	73.088%	0.0447	0.0130	0.0163
3	7.049	1.829	74.053%	1.631	76.856%	0.0259	0.0216	0.0175
4	5.420	2.284	57.864%	1.301	76.001%	0.0494	0.0333	0.0539
5	5.148	1.765	65.722%	1.261	75.497%	0.0349	0.0419	0.0163
6	13.86 7	5.377	61.226%	3.796	72.628%	0.0426	0.0341	0.0204
7	16.18 4	7.273	55.060%	4.172	74.220%	0.0747	0.0299	0.0240
8	16.96 1	8.669	48.892%	4.534	73.267%	0.1051	0.0494	0.0643
9	7.843	2.074	73.561%	2.185	72.134%	0.1737	0.1390	0.0356
10	6.719	2.122	68.418%	2.329	65.341%	0.0461	0.0184	0.0286
11	6.523	1.499	77.026%	1.380	78.852%	0.0367	0.0040	0.0329
12	5.252	1.978	62.347%	1.622	69.113%	0.0633	0.0456	0.0279
13	7.673	3.209	58.181%	2.193	71.423%	0.0567	0.0402	0.0257
14	5.414	2.428	55.154%	1.360	74.891%	0.0358	0.0125	0.0475
15	6.905	2.287	66.884%	1.532	77.811%	0.0593	0.0196	0.0180
avg			64.808%		74.04%	0.058	0.034	0.03

The average saving voting time by PPHT is about 64.8%; the saving by improved Hough is about 74.04%. Improved Hough has slightly a higher performance than PPHT. Both PPHT and improved Hough have dramatic improvement compared with SHT. The result proved that both PPHT and improved Hough have the ability to minimize the number of voting operations. The accurate rate evaluation between the improved Hough and PPHT is illustrated in Table 4.7.

**Table 4.7 The accurate rate of the Hough algorithms**

	<b>Correct - detection (Frames)</b>	<b>Total Frames</b>	<b>total time /s</b>	<b>average time ms/f</b>	<b>Detection rate</b>
PPHT	10884	12000	432	36	90.70%
Improved Hough	11439	12000	384	32	95.33%

The improved Hough process time on average is 32 ms/f, the standard camera frame rate in real time is 35.7 ms/f, namely 28f/s, the processing speed can catch up with the frame refresh speed. In every second the algorithm can deal with 31.25 frames, faster than the speed of camera capture 28f/s. If the car is running at a speed of 120km/h that means the car moves approximately 1.06 m per frame process time which is able to meet the real time requirement. The PPHT process in every frame is about 1.2m. The PPHT processing time is longer than the Improved Hough and miss-detection rate is also higher than improved Hough.

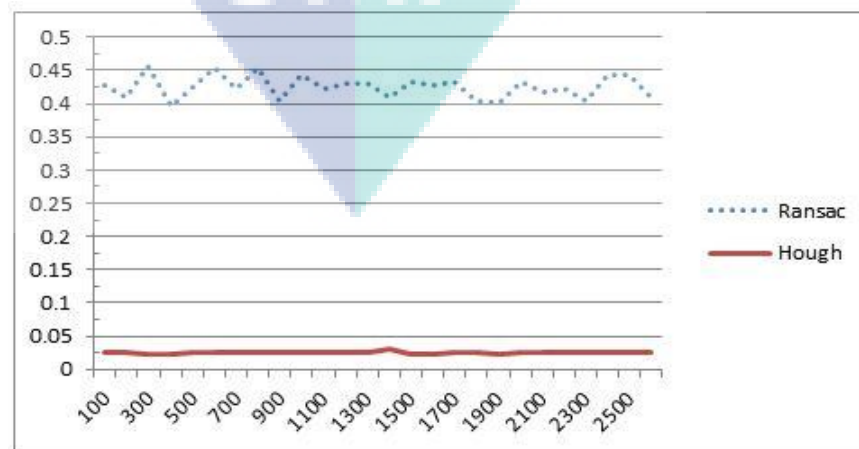
#### **4.6 PERFORMANCE EVALUATION OF IMPROVED HOUGH AND RANSAC**

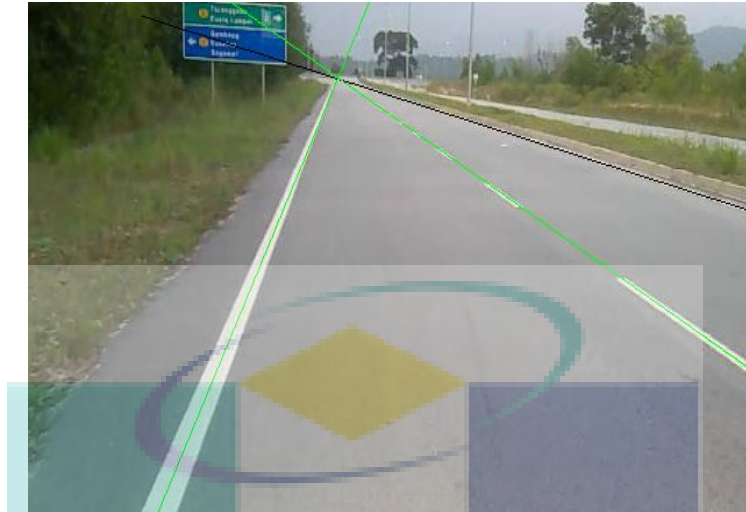
Performance evaluation and analysis about improved Hough and RANSAC are briefly explained in this section. The examination between those two algorithms is categorized in three viewpoints: accuracy, computing time, and robustness. Table 4.8 and Figure 4.12 below show the correct rate and computational time of each image sequence.

**Table 4.8** Accuracy rates of the Lane Detection Technologies

Frames	#Detected		# Errors		# Correct rate	
	Hough	RANSAC	Hough	RANSAC	Hough	RANSAC
1-300	289	246	11	54	96.33%	82.00%
301-600	291	266	9	34	97.00%	88.67%
601-900	283	257	17	43	94.33%	85.67%
901-1200	288	235	12	65	96.00%	78.33%
1201-1500	291	217	9	83	97.00%	72.33%
1501-1800	284	245	16	55	94.67%	81.67%
1801-2100	280	260	20	40	93.33%	86.67%
Sum/Average	2006	1726	94	374	95.52%	82.19%

Through the simulation result comparison, we discovered that Hough was significantly more accurate and the processing time was much shorter than RANSAC algorithm. With reference to RANSAC's random nature, each executing time is based on the same image which keeps changing the amount of time in the process. While Hough's Transform processing time is stable, it does not show much change in the running time.

**Figure 4.12** Computational performance in the experiment



**Figure 4.13** Detection effect implemented by RANSAC algorithm

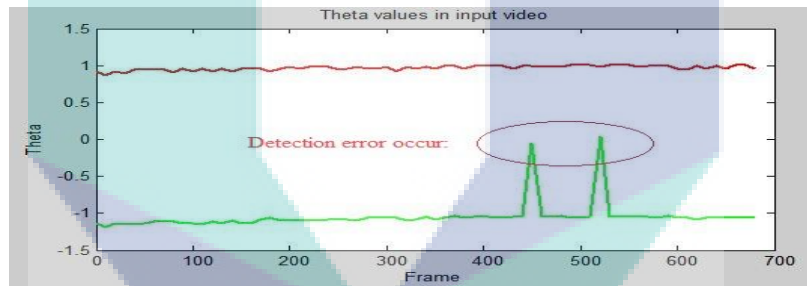
Furthermore RANSAC is heavily depended on the image quality. It should be noted that when there are more pixels passed into the RANSAC algorithm, more time is needed to complete the process. The process speed of the image is not stable as indicating at Figure 4.12; therefore it will slow down the high performance of detection task. An example of fitting of lines by RANSAC algorithm is given in Figure 4.13.

The detection result works under independent conditions is relied on many factors, including the usage of parameters in the program, the collected experimental video and others variables involved in this experiment. Those parameters used in program can produce varying degrees of impact to Hough and RANSAC which would probably make the result more or less accurate or cause changes in the computation time. Moreover those collected experimental video with different image quality and experimental conditions might also lead to inconstant results.

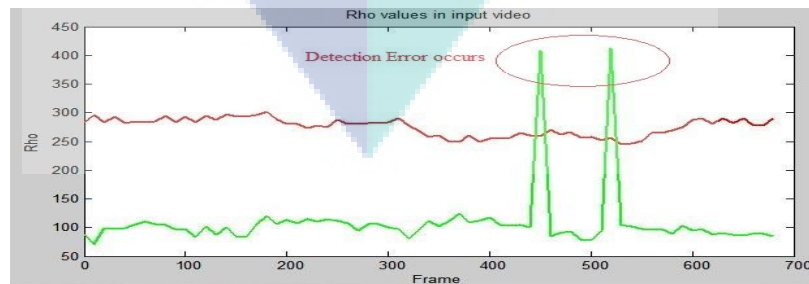
#### 4.7 EXPERIMENT ON DETECTION ERROR APPEARANCE

Since lane markings have consistent and continuous features in the consecutive frames, any lane markings suddenly jump or miss will be considered as a detection error appearance, seen in given examples Figure 4.14, Figure 4.15 and Figure 4.16. The outcomes are recorded and validated automatically in two aspects:

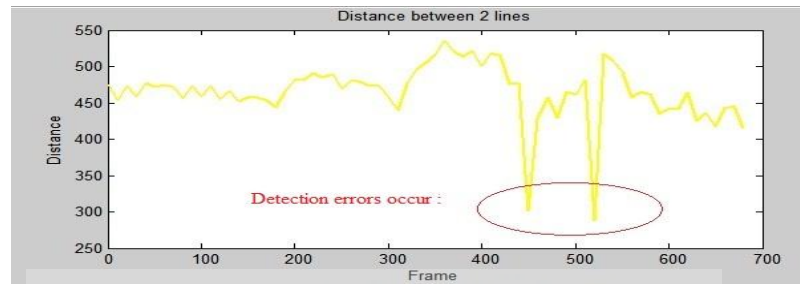
- (i). There is a trace difference in between right and left lane markings. If the distance is changed more than the normal level, it will be considered as an error appearance.
- (ii). An error occurs when fluctuation of Rho and Theta value is out of range in the single side of lane markings.



**Figure 4.14** Detection errors recorded based on Theta value

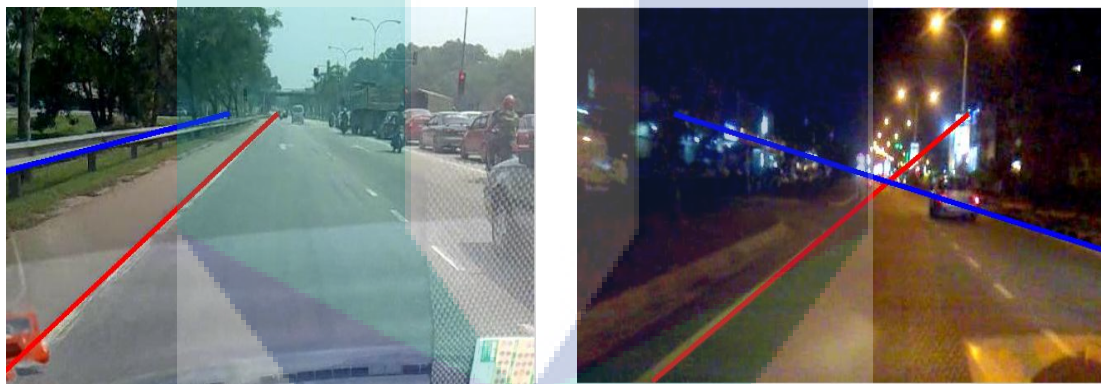


**Figure 4.15** Detection errors recorded based on Rho value



**Figure 4.16** Detection errors recorded based on distance

Two miss-detection examples are given in Figure 4.17. They were caused by failure in the linear feature extraction or blur lane markings with low illumination.

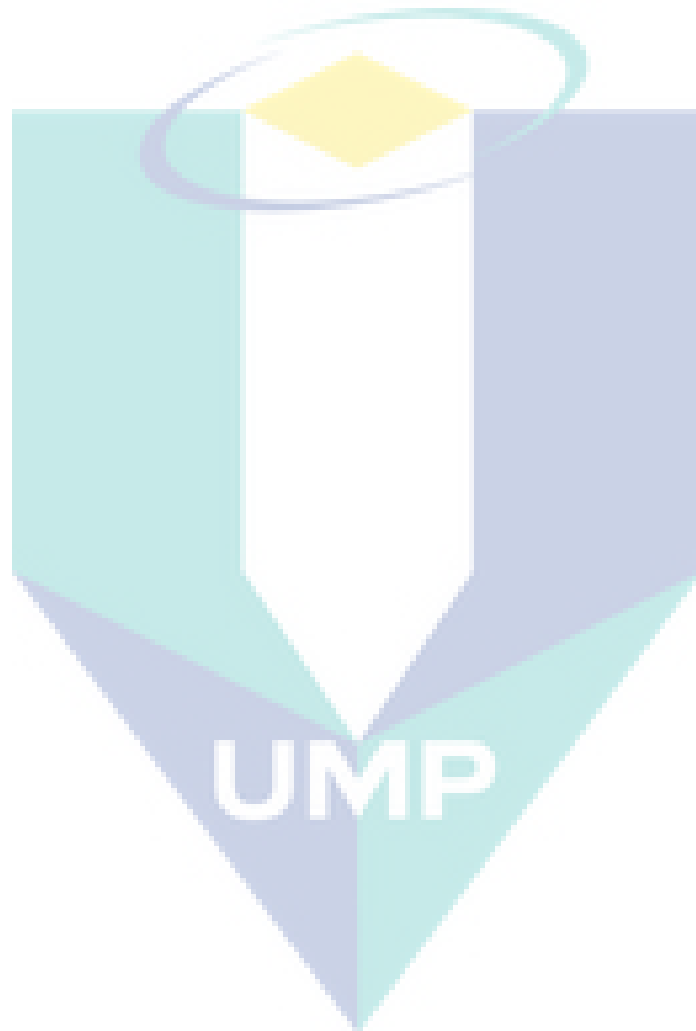


**Figure 4.17** miss-detection frames

## 4.8 CONCLUSION

This chapter presents the experiments and result of analysis. Experiments proved that E-MAMIMA is able to enhance detection result remarkably; minimize negative influence form image noise. The Sobel operator has been chosen as an appropriate operator for lane detection purpose because it performs well in both the computation cost and moderate noise sensitivity. The comparative result between standard Hough, PPHT and improved Hough demonstrated that improved Hough has higher performance than the other two algorithms. Through the simulation result and the comparison of accuracy, computing

time, and robustness between improved Hough and RANSAC, it is clearly identified that Hough significantly is more accurate and the processing time is much shorter than RANSAC algorithm. The improved Hough outperforms the other techniques.





## CHAPTER 5

### CONCLUSION

After the review and analysis of previous literatures and comparison of many related systems, depending on the attributes of structured road and actual needs of high speed car's safety, a lane detection algorithm was proposed to meet the real-time demand, effectiveness, low computation cost and robustness.

The experiments are carried out to compare the lane detection algorithm under different environmental conditions. Both visualized and numeric evaluations have been performed. Overall the combination of E-MAXIMA transformation with the improved Hough proves to be able to accurately distinguish, detect and track the lane markings. The experimental results demonstrate the robustness of the algorithm used. Even under inconsistent illumination and a diversity of road conditions, the lane markings that are disconnected can still be well recognized.

The improved Hough process time on average is 32 ms/f, which means that 31.25 frames is processed in every second for a car running at a speed of 120 km/h. While each frame is being processed, the car would have moved approximately 1.2 m. This processing speed approximates the real-time requirement, thus greatly improves the accuracy and efficiency of the algorithm.

The major achievement of this thesis is: reducing Hough searching space, decreasing voting times about 74.04%, and using E-MAXIMA transformation contributed to lane marking extraction and noise degradation. This algorithm is based on the features of the structured road. The near field-of-view is defined as a straight line model. The lane markings are detected by searching the optimal parameters of the defined lane model. The

PPHT and RANSAC are implemented as alternative algorithms for lane detection function. Improved Hough, PPHT and RANSAC have been evaluated on real world driving data.

The main contents of this research are summarized as follows:

- (i). Objective to seek a more effective image pre-processing method by means of extracting road image features. The experiments for a large number of images have been carried out by selecting and designing high efficiency pre-processing method.
- (ii). The recent research was reviewed and analyzed in detail. Numerous and various lane models have been chosen. Most of them are either too complicated to meet high operating efficiency or work on straight lane only. This thesis discussed the possibility of adopting straight line on curve highway.
- (iii). Image is divided into two parts, near field-of-view and far field-of-view. ROI concentrates on near field-of-view, which simplifies the image background and applies to different road conditions.
- (iv). This thesis reduced the Hough search space resulting in improving the voting scheme. The proposed approach not only significantly improves the performance of the voting scheme, but also produces a cleaner voting map and makes the transform more robust.

This work may apply to various applications of autonomous driving. It proposed a strategy to design a human-vehicle interface that is aware of the drivers' current needs and capabilities. Applied to the design of driver assistance system, this could lead to the development of safer vehicles. Avoiding dangerous situations, when the driver lose focus during driving, it can help build up the trustful relationship between the driver and the vehicle.

Overall the combination of E-MAXIMA transformation with improved Hough proved to be able to accurately distinguish, detect and track the desired road lane markings.

The proposed system is suitable for driver training, driving security assistance and autonomous navigation.

## 5.1 FUTURE WORK

The results presented in the previous chapter are promising; however, lots of enhancements in many aspects are still needed that this research could borrow in the future.

In order to meet real-time request and reduce the computation burden, this system is based on the most simple straight line model. In reality, the road conditions are much more complex, many actual situations do not cover, for example, spiral roads and rural roads. Lane detection algorithm should be used to handle and describe those cases.

In snowy weather, road surface is easily covered which will cause a strong interference in image resulting in the failure of detecting the algorithm. The image pre-processing algorithm needs more optimization in order to increase the precision of detection process.

This lane detection system designs in a hypothetical situation that roads are even where uphill and downhill roads do not list as one of the consideration factors. It still needs in-depth study about the geometric relation of road and digital camera imaging.

A further development is to link the system with the vehicles turning signal so when the driver decides to exit the road or make a U turn, the system will be able to detect a driver controlled departure and not give a false alarm. When the driver deviates out of the road without a turning signal, the system would be able to make more intense noticeable alarm.

Another future work can be done to expand this system to FPGA controller or other micro controller which would be able to embed this into a portable hardware.

## REFERENCES

- Accidents cost Malaysia RM9.3bil. (Malaysia Road Safety Department, 2010), from <http://thestar.com.my/news/story.asp?sec=nation&file=/2010/8/25/nation/20100825140400>
- Ali, M., & Clausi, D. (2001). *Using the canny edge detector for feature extraction and enhancement of remote sensing images*. In IGARSS-01, IEEE international symposium on geoscience and remote sensing (pp.2298–2300), Sydney, Australia.
- Amo, M., Martinez, F., & Torre, M. (2006). Road extraction from aerial images using a region competition algorithm. *Image Processing, IEEE Transactions on Image Processing*, 15(5), 1192-1201.
- Bansal, B., Saini, J. S., Bansal, V., & Kaur, G. (2012). Comparison of various edge detection techniques. *Journal of Information and Operations Management*, 3(1, 2012), 103-106.
- Ben Romdhane, N., Hammami, M., & Ben-Abdallah, H. (2011). A comparative study of vision-based lane detection methods. *Advances Concepts for Intelligent Vision Systems*, 46-57.
- Beucher, S., & Bilodeau, M. (1994). *Road segmentation and obstacle detection by a fast watershed transformation*. Paper presented at the Intelligent Vehicles' 94 Symposium, Proceedings of the Computing & Processing (Hardware/Software) Transportation, 149-154.
- Borkar, A., Hayes, M., & Smith, M. T. (2009). *Robust lane detection and tracking with RANSAC and Kalman filter*. Paper presented at the Image Processing (ICIP), 2009 16th IEEE International Conference Center for Signal & Image Process. (CSIP), Georgia Inst. of Technol., Atlanta, GA, USA, 3261 -3264.
- Bowyer, K. W., Chang, K., & Flynn, P. (2006). A survey of approaches and challenges in 3D and multi-modal 3D+ 2D face recognition. *Computer Vision and Image Understanding*, 101(1), 1-15.
- Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6), 679-698.

- Chiu, K. Y., & Lin, S. F. (2005). *Lane detection using color-based segmentation*. Paper presented at the Intelligent Vehicles Symposium, 2005. Proceedings of IEEE, 706-711.
- Cunningham, H., Tablan, V., Roberts, I., Greenwood, M. A., & Aswani, N. (2011). Information Extraction and Semantic Annotation for Multi-Paradigm Information Management. *Current Challenges in Patent Information Retrieval*, 307-327.
- Engel, K., Hadwiger, M., Kniss, J., Rezk-Salama, C., & Weiskopf, D. (2006). *Real-time volume graphics*. Eurographics Association.
- Fan, C., & Ren, Y. (2010). *Study on the Edge Detection Algorithms of Road Image*. Paper presented at the Third International Symposium on Information Processing (ISIP), 217-220.
- Fan, Y., Zhang, W., Li, X., Zhang, L., & Cheng, Z. (2011). *A robust lane boundaries detection algorithm based on gradient distribution features*. Paper presented at the Fuzzy Systems and Knowledge Discovery (FSKD), 2011 Eighth International Conference , 3, 1714-1718.
- Fernandes, L. A. F., & Oliveira, M. M. (2008). Real-time line detection through an improved Hough transform voting scheme. *Pattern recognition*, 41(1), 299-314.
- Gonzalez, R. C., & Woods, R. (2008). *Digital image processing*. 3rd ed. Upper Saddle River (NJ): Prentice Hall; ISBN:9780131687288., 738-741.
- Green, B. (2002). Canny edge detection tutorial. *from web resource*. [www. pages. drexel. edu/weg22/cantut. html](http://www.pages.drexel.edu/weg22/cantut.html).
- A Guide on Geometric Design of Roads, Jabatan Kerja Raya Malaysia. ( Arahan Teknik 8/86), from [http://rakan1.jkr.gov.my/cjalan/editor/files/Guide%20On%20Geometric%20Design%20Of%20Road\(2\).pdf](http://rakan1.jkr.gov.my/cjalan/editor/files/Guide%20On%20Geometric%20Design%20Of%20Road(2).pdf)
- He, W., Wang, X., Chen, G., Guo, M., Zhang, T., Han, P., & Zhang, R. (2011). *Monocular based lane-change on scaled-down autonomous vehicles*. Paper presented at the Intelligent Vehicles Symposium (IV), 2011 IEEE.
- He, Y., Wang, H., & Zhang, B. (2004). Color-based road detection in urban traffic scenes. *Intelligent Transportation Systems, IEEE Transactions on*, 5(4), 309-318.

- Hongjian, S., Ward, R. (2002). *Canny edge based image expansion*, In ISCAS-2002, IEEE international symposium on circuits and systems (pp. 785–788). Scottsdale, AZ, USA.
- Illingworth, J., & Kittler, J. (1988). A survey of the Hough transform. *Computer vision, graphics, and image processing*, 44(1), 87-116.
- Jau, U. L., Teh, C. S., & Ng, G. W. (2008). *A comparison of RGB and HSI color segmentation in real-time video images: A preliminary study on road sign detection*. Paper presented at the Information Technology, 2008. ITSIM 2008. International Symposium ,4, 1-6.
- Jeong, P., & Nedeveschi, S. (2005). Efficient and robust classification method using combined feature vector for lane detection. *Circuits and Systems for Video Technology, IEEE Transactions*, 4, 1-6.
- Jung, C. R., & Kelber, C. R. (2004). *A lane departure warning system based on a linear-parabolic lane model*. Paper presented at the Intelligent Vehicles Symposium, 2004 IEEE, 891 – 895.
- Kaliyaperumal, K., Lakshmanan, S., & Kluge, K. (2001). An algorithm for detecting roads and obstacles in radar images. *Vehicular Technology, IEEE Transactions*, 50(1), 170-182.
- Kang, Y., Kidono, K., Naito, T., & Ninomiya, Y. (2008). *Multiband image segmentation and object recognition using texture filter banks*. Paper presented at Pattern Recognition, 2008. ICPR 2008. 19th International Conference, 1-4.
- Kastrinaki, V., Zervakis, M., & Kalaitzakis, K. (2003). A survey of video processing techniques for traffic applications. *Image and Vision Computing*, 21(4), 359-381.
- Kim, Z. W. (2008). Robust lane detection and tracking in challenging scenarios. *Intelligent Transportation Systems, IEEE Transactions on*, 9(1), 16-26.
- Kirsch, R. A. (1971). Computer determination of the constituent structure of biological images. *Computers and Biomedical Research*, 4, 314–328.
- Kiryati, N., Eldar, Y., & Bruckstein, A. M. (1991). A probabilistic Hough transform. *Pattern recognition*, 24(4), 303-316.

- Kreucher, C., Lakshmanan, S., & Kluge, K. (1998). *A driver warning system based on the LOIS lane detection algorithm*. Paper presented at Proc. IEEE Int.Conf.Intelligent Vehicles, Stuttgart, Germany, 17–22.
- Li, Q., Zheng, N., & Cheng, H. (2004). Springrobot: A prototype autonomous vehicle and its algorithms for lane detection. *Intelligent Transportation Systems, IEEE Transactions on*, 5(4), 300-308.
- Li, Z., Cai, Z., Xie, J., & Ren, X. (2012). *Road markings extraction based on threshold segmentation*. Paper presented at the Fuzzy Systems and Knowledge Discovery (FSKD), 2012 9th International Conference , 1924-1928.
- Lipski, C., Scholz, B., Berger, K., Linz, C., Stich, T., & Magnor, M. (2008). *A fast and robust approach to lane marking detection and lane tracking*. Paper presented at Image Analysis and Interpretation, 2008. SSIAI 2008. IEEE Southwest Symposium , 57-60.
- Liu, X., Wang, G., Liao, J., Li, B., He, Q., & Meng, M. Q. H. (2012). *Detection of geometric shape for traffic lane and mark*. Paper presented at the Information and Automation (ICIA), 2012 International Conference, 395-399.
- López, A., Canero, C., Serrat, J., Saludes, J., Lumbreras, F., & Graf, T. (2005). *Detection of Lane Markings based on Ridgeness and RANSAC*. Paper presented at the Intelligent Transportation Systems, 2005. Proceedings of 2005 IEEE, 254-259.
- Lu, W., & Tan, J. (2008). Detection of incomplete ellipse in images with strong noise by iterative randomized Hough transform (IRHT). *Pattern recognition*, 41(4), 1268-1279.
- Mahbub, U., Imtiaz, H., & Rahman Ahad, M. (2011). *An optical flow based approach for action recognition*. Paper presented at the Computer and Information Technology (ICCIT), 2011, 14th International Conference, 646-651.
- Maitra, I. K., Nag, S., & Bandyopadhyay, S. K. (2012). A Novel Edge Detection Algorithm for Digital Mammogram. *International Journal of Information*, 2(2).
- MUFORS;. (2011). Malaysia Sees Increased Road Fatalities, from <http://www.roadtraffic-technology.com/news/news108439.html>
- MUTCD. (2009). Manual on Uniform Traffic Control Devices from <http://mutcd.fhwa.dot.gov/>



- Nguyen, T. T., Dai Pham, X., & Jeon, J. W. (2008). *An improvement of the Standard Hough Transform to detect line segments*. Paper presented at the Industrial Technology, 2008. ICIT 2008. IEEE International Conference, 1-6.
- Nixon, M. S., & Aguado, A. S. (2008). *Feature extraction and image processing*: Academic Press.
- Obradović, Đ., Konjović, Z., Pap, E., & Rudas, I. (2012). Linear fuzzy space based road lane model and detection. *Knowledge-Based Systems*, 37-47.
- Pande, S., Bhadouria, V. S., & Ghoshal, D. (2012). A Study on Edge Marking Scheme of Various Standard Edge Detectors. *International Journal of Computer Applications*, 44(9), 33-37.
- Pomerleau, D. (1995). *RALPH: Rapidly adapting lateral position handler*. Intelligent Vehicles '95 Symposium, 506-511.
- Prewitt, J. M. S. (1970). *Object enhancement and extraction, picture processing and psychopictorics*. New York: Academic Press.
- Rafael, C., Gonzalez, J. P., Richard, E. W., & Steven L, E. (2004). *Digital Image Processing Using MATLAB* (pp4-5).
- Rios Cabrera, R., Tuytelaars, T., & Van Gool, L. (2011). *Efficient Multi-Camera Detection, Tracking, and Identification using a Shared Set of Haar-Features*. Paper presented at the Proceedings IEEE computer society conference on computer vision and pattern recognition-CVPR2011.
- Rothschild, L. M. (2012). *System and method of integrating lane position monitoring with locational information systems*: US Patent 20,120,109,521.
- Schreiber, D., Alefs, B., & Clabian, M. (2005). *Single camera lane detection and tracking*. Paper presented at the Intelligent Transportation Systems, 2005. Proceedings of 2005 IEEE.
- Sehestedt, S., Kodagoda, S., Alempijevic, A., & Dissanayake, G. (2007). *Efficient lane detection and tracking in urban environments*. Paper presented at 3rd European Conference on Mobile Robots (EMCR 07), Freiburg, Germany.



- Sobel, I. (1978). Neighborhood coding of binary images for fast contour following and general binary array processing. *Computer Graphics and Image Processing*, 8(1), 127-135.
- Soille, P. (2004). *Morphological image analysis*. 2nd ed. New York: Springer.
- Song, M., & Civco, D. (2004). Road extraction using SVM and image segmentation. *Photogrammetric engineering and remote sensing*, 70(12), 1365-1371.
- Sun, T. Y., Tsai, S. J., & Chan, V. (2006). *HSI color model based Lane-Marking detection*. Paper presented at the 2006 IEEE Intelligent Transportation Systems Conference.
- Tan, C., Hong, T., Chang, T., & Shneier, M. (2006). *Color model-based real-time learning for road following*. Paper presented at the Intelligent Transportation Systems Conference, 2006. ITSC'06 IEEE.
- The HSI color space. (2011), from <http://www.blackice.com/colorspaceHSI.htm>
- Truong, Q. B., Lee, B. R., Heo, N. G., Yum, Y. J., & Kim, J. G. (2008). *Lane boundaries detection algorithm using vector lane concept*. Paper presented Control, Automation, Robotics and Vision, 2008. ICARCV 2008. 10th International Conference, 2319-2325
- Wang, H., & Shao, S. L. (2011). Lane Markers Detection Based on Consecutive Threshold Segmentation. *Advanced Materials Research*, 317, 881-885.
- Wang, J., Gu, F., Zhang, C., & Zhang, G. (2010). *Lane boundary detection based on parabola model*. Paper presented at the Information and Automation (ICIA), 2010 IEEE International Conference, 1729 -1734.
- Wang, J., Wu, Y., Liang, Z., & Xi, Y. (2010). *Lane detection based on random hough transform on region of interesting*. Paper presented at the Information and Automation (ICIA), 2010 IEEE International Conference, 1735- 1740
- Wang, Y., Teoh, E. K., & Shen, D. (2004). Lane detection and tracking using B-Snake. *Image and Vision Computing*, 22(4), 269-280. doi: 10.1016/j.imavis.2003.10.003
- Wu, B. F., Chen, W. H., Chang, C. W., Chen, C. J., & Chung, M. W. (2007). *A new vehicle detection with distance estimation for lane change warning systems*. Paper presented at Intelligent Vehicles Symposium, 2007 IEEE, 698-703

- Xu, L., Oja, E., & Kultanen, P. (1990). A new curve detection method: randomized Hough transform (RHT). *Pattern Recognition Letters*, 11(5), 331-338.
- Yu, B., & Jain, A. K. (1997). *Lane boundary detection using a multiresolution hough transform*. Paper presented at Image Processing, 1997. Proceedings., International Conference on ,2,748-751
- Yu, X., Beucher, S., & Bilodeau, M. (1992). *Road tracking, lane segmentation and obstacle recognition by mathematical morphology*. Paper presented at the Intelligent Vehicles' 92 Symposium., 166-172.
- Yuksel, M. E. (2007). Edge detection in noisy images by neuro-fuzzy processing. *International Journal of Electronics and Communications (AEU)*, 61(2), 82-89.
- Yuen, H., Princen, J., Illingworth, J., & Kittler, J. (1990). Comparative study of Hough transform methods for circle finding. *Image and Vision Computing*, 8(1), 71-77.
- Zhai, L., Dong, S., & Ma, H. (2008). *Recent methods and applications on image edge detection*. Paper presented at the Education Technology and Training, 2008.
- Zhang, L., & Wu, E. (2009). *A Road Segmentation and Road Type Identification Approach Based on New-Type Histogram Calculation*. Paper presented at the Image and Signal Processing, 2009. CISP'09. 2nd International Congress ,1-5.

### LIST OF PUBLICATIONS

1. Kamarul Hawari Bin Ghazali, Jie Ma, Rui Xiao, Solly Aryza lubis, " An Innovative Face Detection Based on YCgCr Color Space", International Conference On Broadcast Technology and Multimedia Communication, chongqing,China, December 13-14,2010,Vol.5,pp.145-148.
2. Kamarul Hawari bin Ghazali, Rui Xiao, Jie Ma,"An intelligent Lane markers recognition and localization system using improved Hough Transform", Applied Mechanics and Materials(1660-9336), Vols. 121-126(2012), pp.1186-1190.
3. Kamarul Hawari bin Ghazali, Jie Ma, Rui Xiao, "Multi-angle Face Detection Using Back Propagation Neural Network", Applied Mechanics and Materials(1660-9336), Vols. 121-126(2012), pp.2411-2415.
4. Kamarul Hawari bin Ghazali, Jie Ma, Rui Xiao, "An Innovative Face Detection based on Skin Color Segmentation", International Journal of Computer Applications(0975-8887), Volume 34-No.2, November 2011,pp.6-10.
5. Kamarul Hawari bin Ghazali, Jie Ma, Rui Xiao,"Driver's Face Tracking Based on Improved CAMShift", International Journal of Image, Graphics and Signal Processing (2074-9082), 2013, vol.1, pp.1-7.
6. Kamarul Hawari bin Ghazali, Jie Ma, Rui Xiao,"A PERCLOS-based Driver Fatigue Detection", The First International Conference on Information Science and Management, Toba Lake, North Sumatra, Indonesia, December 3-5, 2012.
7. Kamarul Hawari bin Ghazali, Rui Xiao,Jie Ma, "A Lane Marker Detection system based on Improved Hough Transform", The First International Conference on Information Science and Management, Toba Lake, North Sumatra, Indonesia, December 3-5, 2012.
8. Jie Ma, Rui Xiao,Kamarul Hawari bin Ghazali,"Driver's Face Tracking Based On Improved CAMShift for Drowsiness Detection", Applied Math. & Information Sciences Letters, No.1, pp.31-34(2012).
9. Kamarul Hawari bin Ghazali, Rui Xiao,Jie Ma," Road Lane Detection Using H-Maxima And Improved Hough Transform", The Fourth International Conference On Computational Intelligence, Modelling and Simulation, Kuantan, Malaysia, September 25-27, 2012.